PCI Hot Plug

User's Guide I/O device edition

Preface

Purpose

This manual describes the procedure of exchanging or extending file PCI cards and network PCI cards using the PCI Hot Plug feature.

Intended Readers

This manual is intended for the following readers:

- System software developers and test engineers
- System support engineers
- System administrators who uses this function
- Hardware engineers who uses this function

Organization

This manual is organized as follows:

- "Chapter 1 PCI cards with support for PCI Hot Plug" describes the lists of PCI cards supporting PCI Hot Plug.
- "Chapter 2 Overview of replacement and extension of PCI cards" describes the outline of a procedure of replacement and extension of PCI cards.
- "Chapter 3 File devices" describes the detailed procedure of PCI Hot Plug about file devices.
- "Chapter 4 Network devices" describes the detailed procedure of PCI Hot Plug about network devices.
- "Chapter 5 Notes for PCI Hot Plug" describes the notes when using PCI Hot Plug.

Notation Used

The following shows the notation conventions used in this manual.

- In this manual, the Solaris(TM) Operating Environment is abbreviated to Solaris OE, the Solaris(TM) 8 Operating Environment is abbreviated to Solaris 8 OE, the Solaris(TM) 9 Operating Environment is abbreviated to Solaris 9 OE.
- The titles of chapters are enclosed in parentheses ("").
  Example: See "Chapter1 PCI cards supporting PCI Hot Plug "
- Commands and other input use the following prompts:
  C shell prompt:
  prompt%
  Bourne and korn shell prompt:
  prompt$
  Super user prompt:
  #
- Commands entered by the user is shown in bold:
  # cfgadm -c configure pcipsy2:R0B00-PCI#slot01<Return>
- Key combinations are represented, for example, by CTL-C, which means to simultaneously press the Control and C key.

Reference Manuals

If necessary, refer to the following manuals:

- PCI Hot Plug User's Guide
- Dynamic Reconfiguration User's Guide

Trademarks

Sun, Sun Microsystems, the Sun Logo, Solaris and all Solaris based marks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries, and are used under license.

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products with the SPARC trademark are based on architecture developed by Sun Microsystems.

All other products, or company names mentioned in this document are claimed as trademark and trade names by their respective companies.

FUJITSU LIMITED
April, 2003

January 2003 First Edition
April 2003 Second Edition

Notes

- This manual may not be copied by any means without the expressed written permission of FUJITSU LIMITED.
- FUJITSU LIMITED reserves the right to make changes to this manual without prior notice.

Copyright 1995- GLOBEtrotter Software, Inc.
Copyright 1995- Software Research Associates, Inc.
All Rights Reserved.

All Rights Reserved, Copyright (C) FUJITSU LIMITED 2003

Contents

Chapter 1 PCI cards with support for PCI Hot Plug 1
1.1 PCI cards for file devices 1
1.2 PCI cards for network devices 2
Chapter 2 Overview of replacement and extension of PCI cards 3
2.1 Replacement of PCI card 3
2.1.1 Replacement of PCI cards on non-redundant system 3
2.1.2 Replacement of PCI cards on redundant system 3
2.2 Extension of PCI cards 4
2.2.1 Extension of PCI cards on non-redundant system 4
2.2.2 Extension of PCI cards with new redundant system 4
2.2.3 Extension of PCI cards to redundant system 5
Chapter 3 File devices 7
3.1 Replacement of PCI card 7
3.1.1 Replacement of PCI cards on non-redundant system 7
3.1.2 Replacement of PCI cards on redundant system 12
3.2 Extension of PCI cards 15
3.2.1 Solaris 8 OE 16
3.2.1.1 Extension of PCI cards on non-redundant system 16
3.2.1.2 Extension of PCI cards on new redundant system 25
3.2.1.3 Extension of PCI cards to redundant system 33
3.2.2 Solaris 9 OE 40
3.2.2.1 Extension of PCI cards on non-redundant system 40
3.2.2.2 Extension of PCI cards with new redundant system 49
3.2.2.3 Extension of PCI cards to redundant system 57
Chapter 4 Network devices 65
4.1 Replacement of PCI card 65
4.1.1 Replacement of PCI cards on non-redundant system 65
4.1.2 Replacement of PCI cards on redundant system 67
4.2 Extension of PCI cards 69
4.2.1 Extension of PCI cards on non-redundant system 69
4.2.2 Extension of PCI cards with new redundant system 71
4.2.3 Extension of PCI cards to redundant system 73
Chapter 5 Notes, Restrictions 77
5.1 Notes 77
5.1.1 Network driver configuration changing 77
5.1.2 Using tape device while PCI Hot Plug operation 78
5.1.3 Others 78
5.2 Restrictions 78

Chapter 1 PCI cards with support for PCI Hot Plug

This chapter shows the list of PCI cards supporting PCI Hot Plug feature.

The driver patches and the option driver packages enclosed with a PCI card in this chapter can be downloaded from the following URLs.

- The driver patches
  http://software.fujitsu.com/en/security/index.html
- The option driver packages enclosed with a PCI card
  http://primepower.fujitsu.com/en/driver/e_index.html

Please contact our customer service about the latest information of the driver patches and the option driver packages.

1.1 PCI cards for file devices

The following list describes the PCI cards that support PCI Hot Plug for file devices. Please ask the sales staff about the latest information of PCI Hot Plug supporting cards.

Functions	Product ID	Software of driver (driver name)	Operating clock (*1)	Supporting Functions		Redundant Software
				Hot replacement	Hot expansion	
SCSI	X6540A-U	Solaris OE (glm) **(*2)**	33MHz	NG **(*3)**	NG **(*3)**	none
	X6541A-U X6541A-X	Solaris OE (glm) **(*2)**	33MHz	NG **(*4)**	OK	·Multipath Disk Control 2.2.1 or later **(*5)** ·GR Multipath Driver 1.0.1 or later **(*6)**
	GP7B8SC1U GP7B8SC1X	Solaris OE (glm) **(*2)**	33MHz	OK	OK	·Multipath Disk Control 2.2.1 or later **(*5)** ·GR Multipath Driver 1.0.1 or later **(*6)**
	PP028SC1U PP028SC1X	FUJITSU PCI SCSI Adapter Driver 1.0 or later (enclosed with a PCI card)	33MHz	OK	OK	none
	PW008SC3	FUJITSU ULTRA LVD SCSI Host Bus Adapter Driver 1.0 or later (enclosed with a PCI card)	33MHz / 66MHz	NG **(*3)**	NG **(*3)**	none
Fibre Channel	GP7B8FC1U GP7B8FC1X	FUJITSU PCI Fibre Channel 2.2 or later (enclosed with a PCI card) **(*7)(*8)**	33MHz	OK	OK	·Multipath Disk Control 2.2.1 or later **(*5)** ·GR Multipath Driver 1.0.1 or later **(*6)**
	PW008FC2U	FUJITSU PCI Fibre Channel 2.2.1 or later (enclosed with a PCI card)	33MHz / 66MHz	OK	OK	·Multipath Disk Control 2.2.1 or later **(*5)** ·GR Multipath Driver 1.0.1 or later **(*6)**

April, 2003

(*1) Please refer to "2.1.1.2 PCI Hot Plug configure operation" of the "PCI Hot Plug User's Guide" about the operating clock of PCI cards and PCI slots.
(*2) In case of using Solaris 8 OE, patch 109885-09 or later is needed.
(*3) Replacement and expansion are not supported due to hardware specifications.
(*4) Replacement is not supported due to hardware specifications.
(*5) Multipath Disk Control load balance option is not supporting PCI Hot Plug.
(*6) Please contact our customer service about support to Solaris 9 OE of GR Multipath Driver.
(*7) If using Solaris 9 OE, it is necessary to apply the "FUJITSU PCI Fibre Channel 2.2 (patch level 08) or later.

(*8) If using FUJITSU PCI Fibre Channel 2.2, patch 910936-08 or later is needed to hot expansion.

1.2 PCI cards for network devices

The following list shows the PCI cards that support PCI Hot Plug for network devices. Please ask the sales staff about the latest information of PCI Hot Plug supporting cards.

Functions	Product ID	Software of driver (driver name)	Operating clock (*1)	Supporting Functions		Redundant Software
				Hot replacement	Hot extension	
Ethernet	X1033A-U X1033A-X	Solaris OE (hme)	33MHz	OK	OK	·SynfinityLink 2.2 **(*2)** ·PRIMECLUSTER GLS 4.0 or later **(*3)**
	PW008FE1	FUJITSU PCI 10/100 Ethernet 1.0 or later (enclosed with a PCI card)	33MHz	OK	OK	·SynfinityLink 2.2 **(*2)** ·PRIMECLUSTER GLS 4.0 or later **(*3)**
	PW008QE1	FUJITSU PCI Quad 10/100 Ethernet 1.0 or later (enclosed with a PCI card) **(*4)**	33MHz	OK	OK	·SynfinityLink 2.2 **(*2)** ·PRIMECLUSTER GLS 4.0 or later **(*3)**
	PW008QE2	FUJITSU PCI Quad 10/100 Ethernet 1.1 or later (enclosed with a PCI card)	33MHz / 66MHz	OK	OK	·SynfinityLink 2.2 **(*2)** ·PRIMECLUSTER GLS 4.0 or later **(*3)**
	PP028GE1U	FUJITSU PCI Gigabit Ethernet 1.4.1 or later (enclosed with a PCI card) **(*5)**	33MHz / 66MHz	OK	OK	·SynfinityLink 2.2 **(*2)** ·PRIMECLUSTER GLS 4.0 or later **(*3)**
	PW008GE1	FUJITSU PCI Gigabit Ethernet 1.4.1 or later (enclosed with a PCI card) **(*5)**	33MHz / 66MHz	OK	OK	·SynfinityLink 2.2 **(*2)** ·PRIMECLUSTER GLS 4.0 or later **(*3)**
	PW008GE4	FUJITSU PCI Gigabit Ethernet 2.0 or later (enclosed with a PCI card)	33MHz / 66MHz	OK	OK	·SynfinityLink 2.2 **(*2)** ·PRIMECLUSTER GLS 4.0 or later **(*3)**
	PW008GE5	FUJITSU PCI Gigabit Ethernet 2.0 or later (enclosed with a PCI card)	33MHz / 66MHz	OK	OK	·SynfinityLink 2.2 **(*2)** ·PRIMECLUSTER GLS 4.0 or later **(*3)**
ATM	GP7B8AT1U GP7B8AT1X	FUJITSU PCI ATM 1.2 or later (enclosed with a PCI card)	33MHz	OK	NG **(*6)**	·SynfinityLink 2.2 **(*2)** ·PRIMECLUSTER GLS 4.0 or later **(*3)**

April, 2003

(*1) Please refer to "2.1.1.2 PCI Hot Plug configure operation" of the "PCI Hot Plug User's Guide" about the operating clock of PCI cards and PCI slots.
(*2) The replacement and expansion are supported only on the Redundant Line Control Function.
(*3) When using the PRIMECLUSTER GLS 4.0, the replacement and expansion are supported only on the Redundant Line Control Function.
(*4) When using the "FUJITSU PCI Quad 10/100 Ethernet 1.0," it is necessary to apply the patch 912216-03 or later. Please contact our customer service about the latest patch.
(*5) If using the "FUJITSU PCI Gigabit Ethernet 1.4 or earlier," it is necessary to apply the driver version 1.4.1 or later. Please contact our customer service about the latest driver.
(*6) System rebooting is needed for hot extension.

Chapter 2 Overview of replacement and extensi on of PCI cards

This chapter describes the outline of a procedure of replacement and extension of PCI cards by using PCI Hot Plug.

2.1 Replacement of PCI card

This section describes outline to exchange PCI cards using PCI Hot Plug feature.

About software which makes PCI card redundantly, please refer to each manual of the software.

2.1.1 Replacement of PCI cards on non-redundant system

Before you exchange PCI cards without redundant software, you must stop applications which using the PCI cards. The replacement procedure of PCI cards is as follows.

Please refer to "Chapter 3 File devices" or "Chapter 4 Network devices" about detailed replacing procedure of PCI cards.

1. Stop the hardware monitoring of the Machine Administration
   Stop the hardware abnormal monitoring of the Machine Administration.
2. Specify the replacing PCI card
   Specify the devices and interfaces that are influenced by replacing PCI cards.
3. Stop applications
   Stop services and high layer products that are influenced by replacing PCI cards.
4. Disconnect the PCI card
   It changes into the state where the PCI card is exchangeable.
5. Replace the PCI card
   Replace the PCI card.
6. Connect the PCI card
   It changes into the state where the PCI card is usable.
7. Start applications
   Start high layer products and services used by the replaced PCI card.
8. Start the hardware monitoring of the Machine Administration
   Update the hardware configuration information, and restart the hardware abnormal monitoring of the Machine Administration.

2.1.2 Replacement of PCI cards on redundant system

When exchanging PCI cards with redundant software, it is possible to exchange PCI cards without stopping services and applications. The replacement procedure of PCI cards is as follows.

Please refer to "Chapter 3 File devices" or "Chapter 4 Network devices" about detailed replacing procedure of PCI cards.

1. Stop the hardware monitoring of the Machine Administration
   Stop the hardware abnormal monitoring of the Machine Administration.
2. Specify the replacing PCI card
   Specify the devices and interfaces that are influenced by replacing PCI cards.
3. Disconnect from redundant system
   Disconnect the devices path and interfaces of PCI cards from the redundant system.
4. Disconnect the PCI card
   It changes into the state where the PCI card is exchangeable.
5. Replace the PCI card
   Replace the PCI card.
6. Connect the PCI card
   It changes into the state where the PCI card is usable.
7. Connect to redundant system
   Connect device paths and interfaces of the replaced PCI card to the redundant system.
8. Switch back the redundant path
   Switch back the online path if needed.
9. Start the hardware monitoring of the Machine Administration
   Update the hardware configuration information, and restart the hardware abnormal monitoring of the Machine Administration.

2.2 Extension of PCI cards

This section describes outline to extend PCI cards using PCI Hot Plug feature.

About software which makes PCI card redundantly, please refer to each manual of the software.

2.2.1 Extension of PCI cards on non-redundant system

When not using the redundant configuration software, the extension procedure of PCI cards is as follows.

Please refer to "Chapter 3 File devices" or "Chapter 4 Network devices" about detailed extending procedure of PCI cards.

1. Setup I/O devices which will be added to system
   Setup devices which will be added to system in the case of file devices on Solaris 8 OE.
2. Stop the hardware monitoring of the Machine Administration
   Stop the hardware abnormal monitoring of the Machine Administration.
3. Add PCI cards
   Extend PCI cards.
4. Connect PCI cards
   It changes into the state where the PCI card is usable.
5. Start the hardware monitoring of the Machine Administration
   Update the hardware configuration information, and restart the hardware abnormal monitoring of the Machine Administration.
6. Connect equipment to PCI cards
   Connect equipment to PCI cards.
7. Set up driver
   a. Add the driver configuration of PCI cards
      Add the driver configuration of PCI cards
   b. Add the configuration of device driver
      Add the configuration of device driver in the case of file devices.
8. Setup applications
   Configure the high layer products and services about extended PCI cards.
9. Start applications
   Start high layer products and services used by the replaced PCI card.

2.2.2 Extension of PCI cards with new redundant system

The procedure of using the extended PCI card and establishing a new redundant configuration is as follows.

Please refer to "Chapter 3 File devices" or "Chapter 4 Network devices" about detailed extending procedure of PCI cards.

1. Setup I/O devices which will be added to system
   Setup devices which will be added to system in the case of file devices on Solaris 8 OE.
2. Stop the hardware monitoring of the Machine Administration
   Stop the hardware abnormal monitoring of the Machine Administration.
3. Add PCI cards
   Extend PCI cards.
4. Connect PCI cards
   It changes into the state where the PCI card is usable.
5. Start the hardware monitoring of the Machine Administration
   Update the hardware configuration information, and restart the hardware abnormal monitoring of the Machine Administration.
6. Connect equipment to PCI cards
   Connect equipment to PCI cards.
7. Setup driver
   a. Add the driver configuration of PCI cards
      Add the driver configuration of PCI cards
   b. Add the configuration of device driver
      Add the configuration of device driver in the case of file devices.
8. Setup redundant system
   Configure the environmental configuration of a new redundant system and activate.
9. Setup applications
   Configure the high layer products and services about a new redundant system.
10. Start applications
    Start high layer products and services used by the new redundant system.

2.2.3 Extension of PCI cards to redundant system

When extending PCI cards to the existing redundant system, it is possible to extend PCI cards without stopping services and applications. The extension procedure of PCI card to existing redundant system is as follows.

1. Setup I/O devices which will be added to system
   Setup devices which will be added to system in the case of file devices on Solaris 8 OE.
2. Stop the hardware monitoring of the Machine Administration
   Stop the hardware abnormal monitoring of the Machine Administration.
3. Add PCI cards
   Extend PCI cards.
4. Connect PCI cards
   It changes into the state where the PCI card is usable.
5. Start the hardware monitoring of the Machine Administration
   Update the hardware configuration information, and restart the hardware abnormal monitoring of the Machine Administration.
6. Connect equipment to PCI cards
   Connect equipment to PCI cards.
7. Set up driver
   a. Add the driver configuration of PCI cards
      Add the driver configuration of PCI cards
   b. Add the configuration of device driver
      Add the configuration of device driver in the case of file devices.
8. Connect to redundant system
   Connect device paths and interfaces of the replaced PCI card to the redundant system.
9. Switch the redundant path
   Switch the online path for extended path if needed.

Chapter 3 File devices

This chapter describes in detail the procedures of replacement and extension of file PCI cards by using PCI Hot Plug.

For details about software to use PCI cards in redundant configuration, see the manuals of each software.

3.1 Replacement of PCI card

This section describes the procedures to exchange PCI cards using the PCI Hot Plug feature on non-redundant and redundant systems.

3.1.1 Replacement of PCI cards on non-redundant system

Before exchanging PCI cards without using redundancy software such as multipath control, applications using the PCI cards must be stopped.

Follow the procedures below to exchange PCI cards on a non-redundant system.

1. Stop the machine administration hardware monitoring daemon
   Execute the following command to stop the hardware monitoring daemon of machine administration.

```
# /usr/sbin/FJSVmadm/prephp  <Return>
```

2. Specify the replacing PCI card
   Follow the instructions below and determine the interface name of the path connecting the target PCI card and I/O devices and the connected I/O device.

   a. Determine from the WARNING messages output on the console the interface name of the path connecting the target PCI card and I/O devices and the connected I/O device.
   In the example below, **glm2** is the interface name of the path connecting the broken PCI card and I/O devices, and **sd20** is the disk device connected to glm2.
   When exchanging a PCI card which has two ports, the physical path name of the connection path between the I/O device corresponding to the other port, and the I/O device connected to that path must be determined in the same manner.
   The physical path name corresponding to the other port can be determined by the fact that the name of the two ports on the same PCI card have a relation like "/pci@89,4000/scsi@2" and "/pci@89,4000/scsi@2,1." The I/O device connected to that path resides under the directory of the physical path name determined.

```
            :
WARNING: /pci@89,4000/scsi@2 (glm2):
         invalid intcode=fe00
            :
WARNING: /pci@89,4000/scsi@2/sd@3,0 (sd20):
         SCSI transport failed: reason 'reset': giving up
            :
```

   The following procedures are also explained assuming the console messages above.

   b. Determine the logical path name under /dev/dsk corresponding to the disk device (sd20) connected to the target PCI card. In the example below, "c2t3d0" is the logical path corresponding to sd20.

```
# ls -l /dev/dsk | grep /pci@89,4000/scsi@2/sd@3,0  <Return>
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t3d0s0
-> ../../devices/pci@89,4000/scsi@2/sd@3,0:a
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t3d0s1
-> ../../devices/pci@89,4000/scsi@2/sd@3,0:b
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t3d0s2
```

```
-> ../../devices/pci@89,4000/scsi@2/sd@3,0:c
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t3d0s3
-> ../../devices/pci@89,4000/scsi@2/sd@3,0:d
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t3d0s4
-> ../../devices/pci@89,4000/scsi@2/sd@3,0:e
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t3d0s5
-> ../../devices/pci@89,4000/scsi@2/sd@3,0:f
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t3d0s6
-> ../../devices/pci@89,4000/scsi@2/sd@3,0:g
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t3d0s7
-> ../../devices/pci@89,4000/scsi@2/sd@3,0:h
```

3. Stop applications
   Stop applications with the following operations.

   a. Stop vold.

```
# sh /etc/init.d/volmgt stop  <Return>
```

   b. Stop the all I/O devices connected to the target PCI card (determined in procedure 2.).

[If the device is a disk unit (file system operation)]

   i. Determine the mountpoint of the disk from the logical path name determined in procedure 2.b. (c2t3d0).

```
# mount | grep c2t3d0  <Return>
/export/home on /dev/dsk/c2t3d0s3 setuid/read/write/largefiles on Mon
Sep 30 01:00:51 2002
/develop/firm on /dev/dsk/c2t3d0s0 setuid/read/write/largefiles on Mon
Sep 30 01:00:51 2002
/develop/drv on /dev/dsk/c2t3d0s1 setuid/read/write/largefiles on Mon
Sep 30 01:00:51 2002
/pub on /dev/dsk/c2t3d0s6 setuid/read/write/largefiles on Mon Sep 30
01:00:50 2002
```

   ii. Stop access to the disk. To check which process is using the target file system, use the fuser(1M) command as follows.

```
# fuser -c /export/home  <Return>
/export/home:   14967c   14571c   14493ctm   14020c   13828tm
13803c   13575c   13133c   13125tm   13107c   12682ctm   12066tm
12048c   11971ctm   11952ctm   11937c   11867c   11846c    349m
```

   iii. Unmount the disk.

```
# umount /export/home  <Return>
# umount /develop/firm  <Return>
# umount /develop/drv  <Return>
# umount /pub  <Return>
```

[If the device is a disk unit (raw access operation) or a tape device]

   i. Check the access statistics of the target disk determined in procedure 2.a.
   Do the same with a tape device.

```
# iostat -xc  <Return>
```

```
                extended device statistics                  cpu
device     r/s  w/s  kr/s  kw/s  wait  actv  svc_t  %w  %b  us  sy  wt
id
sd0       59.7  7.5 474.5  45.0  0.0   3.9  58.6    0  41  3   7  23  67
sd1        0.1  0.3   1.0   2.5  0.0   0.0  16.0    0  0
sd20       0.0  0.1   0.3   0.7  0.0   0.0  14.7    0  0
st82       0.0  0.0   0.0   0.0  0.0   0.0   0.0    0  0
nfs1       0.0  0.0   0.0   0.0  0.0   0.0   0.0    0  0
```

ii. Stop disk access of applications.
Disk access statistics of the target disk can be checked as below.
It is the same with tape devices. The example below shows access statistics of one minute.

```
# sar -d 60 1  <Return>


SunOS machine0 5.8 Generic_108528-05 sun4u   10/02/02


17:56:00  device       %busy   avque   r+w/s  blks/s  avwait  avserv
17:57:00  nfs1            0     0.0     0     0     0.0     0.0
          sd0             2     0.3     2    37     0.0   145.5
          sd0,a           1     0.1     0     5     0.0   301.4
          sd0,b           0     0.0     0     9     0.0    31.5
          sd0,c           0     0.0     0     0     0.0     0.0
          sd0,d           1     0.0     0     4     0.0   126.8
          sd0,e           0     0.0     0     0     0.0     0.0
          sd0,f           1     0.1     1     6     0.0   120.9
          sd0,g           1     0.1     1    14     0.0   111.7
                  :
          sd20            0     0.0     0     0     0.0     0.0
          sd20,a          0     0.0     0     0     0.0     0.0
          sd20,c          0     0.0     0     0     0.0     0.0
          sd20,g          0     0.0     0     0     0.0     0.0
                  :
          st82            0     0.0     0     0     0.0     0.0
                  :
```

iii. Stop (deactivate) applications.
See the manuals of each software for details.

[If the device is a disk unit (swap device)]

i. Show a list of swap devices, and check that the disk of the logical path determined in procedure 2.b. (c2t3d0) is a swap device.

```
# swap -l  <Return>
swapfile              dev    swaplo  blocks  free
/dev/dsk/c2t3d0s4  32,164     16  788384  683680
```

ii. Delete the swap device.

```
# swap -d /dev/dsk/c2t3d0s4  <Return>
```

4. Disconnect the PCI card.
   Disconnect the target PCI card with the following procedure.

   a. By using inst2comp(1M) command, determine the slot position of the PCI card called an attachment point identifier ("Ap_Id") from the interface name of the path connecting the target PCI card and I/O devices determined in procedure 2.a. (glm2).
   Please refer to "PCI Hot Plug User's Guide" about the detail of inst2comp(1M) command.
   In this example, the "Ap_Id" is "**pcipsy3:C0M00-PCI#slot02**"

```
# /usr/sbin/FJSVmadm/inst2comp glm2  <Return>
pcipsy3:C0M00-PCI#slot02
```

   b. Specify "Ap_Id" determined from 4.a. as a parameter and confirm that the slot status of the PCI card to disconnect is "connected configured."

```
# cfgadm pcipsy3:C0M00-PCI#slot02  <Return>
Ap_Id                       Type       Receptacle   Occupant    Condition
pcipsy3:C0M00-PCI#slot02    mult/hp    connected    configured  ok
```

   c. After executing the command to disconnect the PCI card specifying the "Ap_Id" from 4.a., confirm that the slot status has changed to "disconnected unconfigured."

```
# cfgadm -c disconnect pcipsy3:C0M00-PCI#slot02  <Return>
# cfgadm pcipsy3:C0M00-PCI#slot02  <Return>
Ap_Id                       Type       Receptacle    Occupant     Condition
pcipsy3:C0M00-PCI#slot02    unknown    disconnected  unconfigured  unknown
```

   d. To confirm the slot position at replacement operation, blink the ALARM LED of Ap_Id displayed in procedure 4.a.

```
# cfgadm -x led=fault,mode=blink pcipsy3:C0M00-PCI#slot02  <Return>
```

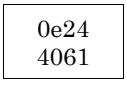
5. Replace the PCI card
   Replace the PCI card disconnected in 4. with a replacement card. This operation is performed by our customer support.
   When exchanging Fibre Channel cards, the following operations are also required.

   [When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS (except for Vixel) and SP5000 SRM Facility]
   Go to procedure 6.

   [When not using SAN management function of the above products]
   To replace PCI cards with the following configurations, Fibre Channel switch SN200 series and disk array device ETERNUS3000/GR700/800 series need to be reconfigured individually.
   - If zone configuration by WWPN(World Wide Port Name) is done on SN200 series.
   - If the Host Affinity function of ETERNUS3000/GR700/800 series is used.

   For details, see "SN200 Series Affinity User's Guide" or "ETERNUS3000/GR700/800 series GRmgr User's Guide."

   To perform the above reconfiguration, the WWPN(a 16-digit number) of the replacement card is needed. The WWPN of the PCI card can be known from the eight characters shown on a label on the front plate of the card. These characters represent the bottom eight digits of the WWPN in hexadecimal form. The top eight digits are fixed to 10000000 in hexadecimal form.
   For example, if the following label is shown on the front plate of the card, the WWPN of the replacement card is 100000000e44061.



Note:
When changing Affinity configuration on SN200 series, I/O to other devices is effected by the change, and may result in temporal errors.
I/O to disk array devices recovers normally because of retry processes, but on Fibre Channel tape devices, backup processes may end in errors. Stop backup before changing Affinity configuration.

6. Connect the PCI card
Connect the replaced PCI card using the cfgadm(1M) command with the configure option, or by pushing the button corresponding to the replacement slot position. Note that the push button is only effective in multiuser mode. After the new PCI card is connected, use the cfgadm(1M) command and confirm that the slot status has changed to "connected configured."
If a large-scale configuration of I/O devices is connected to the PCI card in the target slot, command execution for status confirmation may take time.

```
# cfgadm -c configure pcipsy3:C0M00-PCI#slot02 <Return>
# cfgadm pcipsy3:C0M00-PCI#slot02 <Return>
Ap_Id                      Type       Receptacle  Occupant     Condition
pcipsy3:C0M00-PCI#slot02   mult/hp    connected   configured   ok
```

When exchanging Fibre Channel cards, the following operations are also required.

[When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS (except for Vixel) and SP5000 SRM Facility]
a. Refresh current condition on management window.
SAN (Storage Area Network) management function searchs and indicates replaced PCI card.
This function will adjust switch zoning and host affinity function of disk array device ETERNUS3000/GR700/800 series with new WWN of replaced PCI card.
See “Operation guide” or “User's Manual” of each product for details.

Note:
When changing Affinity configuration on SN200 series, I/O to other devices is effected by the change, and may result in temporal errors.
I/O to disk array devices recovers normally because of retry processes, but on Fibre Channel tape devices, backup processes may end in errors. Stop backup before changing Affinity configuration.

b. Reconnect the replaced PCI card with changed zonig setting by using the following commands.

```
# cfgadm -c disconnect pcipsy3:C0M00-PCI#slot02 <Return>
# cfgadm -c configure pcipsy3:C0M00-PCI#slot02 <Return>
# cfgadm pcipsy3:C0M00-PCI#slot02 <Return>
Ap_Id                      Type       Receptacle    Occupant    Condition
pcipsy3:C0M00-PCI#slot02   mult/hp    connected     configured  ok
```

[When not using SAN management function of the above products]
This procedure is not required.

7. Start applications.
Restart the stopped applications with the following operations.

a. Start vold

```
# sh /etc/init.d/volmgt start <Return>
```

b. Restart the usage of the device stopped in procedure 3.b.

[If the device is a disk unit (file system operation)]
· Mount the unmounted filesystem, and resume usage.

```
# mount /export/home <Return>
# mount /develop/firm <Return>
# mount /develop/drv <Return>
# mount /pub <Return>
```

[If the device is a disk unit (raw access operation) or a tape device]

- Restart applications and resume usage.

Refer to the manual of each application for details.

[If the device is a disk unit (swap device)]

- Add swap device, and resume usage.

```
# /sbin/swapadd -2  <Return>
```

8. Update hardware configuration information of machine administration/ Start the hardware monitoring daemon.
   Execute the following commands to update hardware configuration information of machine administration and to restart the hardware monitoring daemon.

```
# /usr/sbin/FJSVmadm/postphp  <Return>
```

3.1.2 Replacement of PCI cards on redundant system

PCI cards can be replaced without stopping services such as user applications on a redundant system using software such as mutipath control.

Here, the procedure to replace PCI cards on a redundant system using the following redundancy software products is explained.

- Multipath Disk Control (MPHD)
- GR Multipath Driver (GRMPD)

If other redundancy software products are used, see the manual of each product.

1. Stop the machine administration hardware monitoring daemon
   Use the following command to stop the hardware monitoring daemon of machine administration.

```
# /usr/sbin/FJSVmadm/prephp  <Return>
```

2. Specify the replacing PCI card
   Follow the instructions below and determine the interface name of the path connecting the target PCI card and I/O devices and the connected I/O device.

   a. Determine from the WARNING messages output on the console the interface name of the path connecting the target PCI card and I/O devices and the connected I/O device.
   In the example below, **fjpfca3** interface name of the path connecting the target PCI card and I/O devices, and **hddv1**is the LUN(Logical Unit Number) of the disk array device connected to fjpfca3.

```
             :
WARNING: /pci@8d,2000/fibre-channel@1 (fjpfca3):
          Hard Error : PCI DMA error.
             :
WARNING:  /pci@8d,2000/fibre-channel@1/hddv@1,0 (hddv1):
          SCSI transport failed: reason 'reset': giving up
             :
NOTICE: mphd0: I/O path switchover succeeded.
          /pci@8d,2000/fibre-channel@1/hddv@1,0 => /pci@89,2000/fibre-
channel@1/hddv@2,0
             :
```

   The following procedures are also explained assuming the console messages above.

   b. Find the hddv1 with the status "offline fail" from the results of the iompadm command, and determine the logical path name of the LUN connected to the target PCI card.

The following is an example of MPHD, and "**/dev/rdsk/c3t1d0s2**" is the logical path name corresponding to hddv1.
In case of GRMPD, specify "mplb" in the -c option parameter of the iompadm command.

```
# /usr/opt/FJSViomp/bin/iompadm -c mphd -p info  <Return>
              :
IOMP: /dev/FJSVmphd/fiomp/adm2
-> /devices/pseudo/mphd@2:adm
Element:
       /dev/rdsk/c3t1d0s2   offline  fail   block   "target completed hard
reset sequence  [GR7104546-  010000-00-00-30] (hddv1)"
-> /devices/pci@8d,2000/fibre-channel@1/hddv@1,0:c,raw
       /dev/rdsk/c2t2d0s2   online  active   block   "good status with
active  [GR7104546-  010000-01-01-32] (hddv15)"
-> /devices/pci@89,2000/fibre-channel@1/hddv@2,0:c,raw
Node:
       /dev/FJSVmphd/rdsk/mphd2s0
       /dev/FJSVmphd/rdsk/mphd2s1
       /dev/FJSVmphd/rdsk/mphd2s2
       /dev/FJSVmphd/rdsk/mphd2s3
       /dev/FJSVmphd/rdsk/mphd2s4
       /dev/FJSVmphd/rdsk/mphd2s5
       /dev/FJSVmphd/rdsk/mphd2s6
       /dev/FJSVmphd/rdsk/mphd2s7
Function:
       MPmode=false
       AutoPath=true
       Block=true
       NeedSync=false
              :
```

3. Disconnect from redundant system
   To disconnect the path between the target PCI card and I/O devices, execute the following command specifying the logical path name of the connected LUN determined in procedure 2.b. This command only needs to be executed on the representation LUN, and does not need to be executed on each LUN under the same PCI card.
   If the operation path is disconnected on a redundant system, the standby path will automatically be switched to operation.
   If both MPHD/GRMPD controlled devices are connected to the target PCI card, the commands for both products need to be executed.

```
MPHD
# /usr/opt/FJSViomp/bin/iompadm -c mphd change adapter_disconnect /dev/rdsk/c3t1d0s2
<Return>
GRMPD
# /usr/opt/FJSViomp/bin/iompadm -c mplb change adapter_disconnect /dev/rdsk/c3t1d0s2
<Return>
```

4. Disconnect the PCI card.
   Disconnect the defected PCI card with the following procedures.

a. Determine the slot position of the PCI card from the interface name of the path connecting the target PCI card and I/O devices determined in procedure 2.a. (fjpfca3). In this example, the "Ap_Id" is "**pcipsy21:R0B01-PCI#slot03**".

```
# /usr/sbin/FJSVmadm/inst2comp fjpfca3  <Return>
pcipsy21:R0B01-PCI#slot03
```

b. Specify the "Ap_Id" from 4.a. as a parameter and confirm that the slot status of the PCI card to disconnect is "connected configured".

```
# cfgadm pcipsy21:R0B01-PCI#slot03  <Return>
Ap_Id                       Type        Receptacle   Occupant     Condition
pcipsy21:R0B01-PCI#slot03   fibre/hp    connected    configured   ok
```

c. After executing the command to disconnect the PCI card specifying the "Ap_Id" from 4.b., confirm that the slot status has changed to "disconnected unconfigured."

```
# cfgadm -c disconnect pcipsy21:R0B01-PCI#slot03  <Return>
# cfgadm pcipsy21:R0B01-PCI#slot03  <Return>
Ap_Id                        Type       Receptacle    Occupant      Condition
pcipsy21: R0B01-PCI#slot03   unknown    disconnected  unconfigured  unknown
```

d. To confirm the slot position at replacement operation, blink the ALARM LED of Ap_Id displayed in procedure 4.a.

```
# cfgadm -x led=fault,mode=blink pcipsy21:R0B01-PCI#slot03  <Return>
```

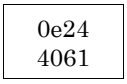
5. Replace the PCI card
Replace the PCI card disconnected in 4. with a replacement card. This operation is performed by our customer support.
When exchanging Fibre Channel cards, the following operations are also required.

[When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS (except for Vixel) and SP5000 SRM Facility]
Go to procedure 6.

[When not using SAN management function of the above products]
To replace PCI cards with the following configurations, Fibre Channel switch SN200 series and disk array device ETERNUS3000/GR700/800 series need to be reconfigured individually.
- If zone configuration by WWPN(World Wide Port Name) is done on SN200 series.
- If the Host Affinity function of ETERNUS3000/GR700/800 series is used.

For details, see "SN200 Series Affinity User's Guide" or "ETERNUS3000/GR700/800 series GRmgr User's Guide."

To perform the above reconfiguration, the WWPN(a 16-digit number) of the replacement card is needed. The WWPN of the PCI card can be known from the eight characters shown on a label on the front plate of the card. These characters represent the bottom eight digits of the WWPN in hexadecimal form. The top eight digits are fixed to 10000000 in hexadecimal form.
For example, if the following label is shown on the front plate of the card, the WWPN of the replacement card is 100000000e44061.



Note:
When changing Affinity configuration on SN200 series, I/O to other devices is effected by the change, and may result in temporal errors.
I/O to disk array devices recovers normally because of retry processes, but on Fibre Channel tape devices, backup processes may end in errors. Stop backup before changing Affinity configuration.

6. Connect the PCI card

Connect the new PCI card with the cfgadm command with the configure option, or by pushing the button corresponding to the replacement slot position. After the new PCI card is connected, confirm that the slot status has changed to "connected configured."

```
# cfgadm -c configure pcipsy21:R0B01-PCI#slot03  <Return>
# cfgadm pcipsy21:R0B01-PCI#slot03  <Return>
Ap_Id                       Type        Receptacle   Occupant     Condition
pcipsy21:R0B01-PCI#slot03   fibre/hp    connected    configured   ok
```

When exchanging Fibre Channel cards, the following operations are also required.

[When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS (except for Vixel) and SP5000 SRM Facility]

a. Refresh current condition on management window.
SAN (Storage Area Network) management function searchs and indicates replaced PCI card.
This function will adjust switch zoning and host affinity function of disk array device ETERNUS3000/GR700/800 series with new WWN of replaced PCI card.
See “Operation guide” or “User's Manual” of each product for details.

Note:
When changing Affinity configuration on SN200 series, I/O to other devices is effected by the change, and may result in temporal errors.
I/O to disk array devices recovers normally because of retry processes, but on Fibre Channel tape devices, backup processes may end in errors. Stop backup before changing Affinity configuration.

b. Reconnect the replaced PCI card with changed zonig setting by using the following commands.

```
# cfgadm -c disconnect pcipsy21:R0B01-PCI#slot03 <Return>
# cfgadm -c configure pcipsy21:R0B01-PCI#slot03 <Return>
# cfgadm pcipsy21:R0B01-PCI#slot03 <Return>
Ap_Id                       Type       Receptacle    Occupant    Condition
pcipsy21:R0B01-PCI#slot03   fibre/hp   connected     configured  ok
```

[When not using SAN management function of the above products]
This procedure is not required.

7. Connect to redundant system
Execute the following command specifying the logical path name from 2.b. to connect to the redundant system. If the operating path switched in procedure 3., the operating path will be automatically switched back.
If both MPHD/GRMPD controlled devices are connected to the PCI card to replace, the commands for both products need to be executed.

```
MPHD
# /usr/opt/FJSViomp/bin/iompadm -c mphd restart adapter_connect /dev/rdsk/c3t1d0s2
<Return>


GRMPD
# /usr/opt/FJSViomp/bin/iompadm -c mplb restart adapter_connect /dev/rdsk/c3t1d0s2
<Return>
```

8. Update hardware configuration information of machine administration/ Start the hardware monitoring daemon.
Execute the following commands to update hardware configuration information of machine administration and to restart the hardware monitoring daemon.

```
# /usr/sbin/FJSVmadm/postphp  <Return>
```

3.2 Extension of PCI cards

This section describes the procedures to add PCI cards using the PCI Hot Plug feature on non-redundant and

redundant systems with Solaris 8 OE and Solaris 9 OE. The difference between the procedures for the two OEs is that I/O devices to be added must be pre-defined before system operation with Solaris 8 OE, but pre-definition is not needed with Solaris 9 OE.

3.2.1 Solaris 8 OE

The procedures to add PCI cards on a Solaris 8 OE system are explained here. To use the PCI Hot Plug function for extension on Solaris OE, I/O devices to be added must be pre-defined and the system must be rebooted before system operation. If the I/O devices to be added are already defined, pre-definition is not needed, and procedure 1. below can be skipped.

3.2.1.1 Extension of PCI cards on non-redundant system

Procedures to add PCI cards without making them redundant on a Solaris 8 OE system are explained here.

1. Setup I/O devices which will be added to system
   Pre-define devices to be added with the following procedures according to the device type. Extension using PCI Hot Plug cannot be done without definition, so devices must be defined before the system goes into operation.

   [Disk/Tape devices(Solaris OE: sd/st)]
   This is an example where a disk device at target=2, lun=0 is pre-defined. If the device is a tape device, re-read "sd" in the following as "st" and perform the same operations.

   a. Confirm that target=2, lun=0 is defined in /kernel/drv/sd.conf.
      If this definition does not exist, add the definition. If it is commented out, remove the comments.

```
Disk device
# vi /kernel/drv/sd.conf  <Return>
          :
name="sd" class="scsi"
     target=0 lun=0;
          :
name="sd" class="scsi"  In case the definition does not exist,
     target=2 lun=0;   add or remove comments.
          :
Tape device
# vi /kernel/drv/st.conf  <Return>
          :
name="st" class="scsi"
     target=0 lun=0;
          :
name="st" class="scsi"  In case the definition does not exist,
     target=5 lun=0;   add or remove comments.
          :
```

   b. In case /kernel/drv/sd.conf was updated in 1.a., reconfigure the kernel , restart the system, and then start system operation.
      If it was not updated, this procedure is not required.

```
# touch /reconfigure  <Return>
# shutdown -y -i6 -g0  <Return>
```

   [Disk Array device(hard disk driver:hddv)]
   This is an example where a disk array device at target=3, lun=0-2 is pre-defined.

If this is the first time to pre-define a disk array device, start from procedure 1.a. If this is not the first time to pre-define a disk array device, start from procedure 1.e.

a. Check if /kernel/drv/hddv.conf exists
   If it does not exist, the hard disk driver is being installed, so start from procedure 1.c. If it does exist, the system is being operated with non-active extension procedures, so start from procedure 1.b.

```
Not exist(new install)      [ to procedure 1.c.]
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf: No such file or directory
Exists(operation with non-active extension procedures) [ to procedure 1.b.]
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf
```

b. When operating with non-active extension procedures, execute the following commands and confirm that /kernel/drv/hddv.conf is deleted.

```
# hddvadm -i  <Return>
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf: No such file or directory
```

c. Use the following command to create an hddv.conf file for the pre-defined extension function.
   The message examples for "operation with non-active extension procedures" means that GR720 currently used with taget=1(lun=0) is registered twice on the sd/hddv driver.

```
New installation            [ to procedure 1.e.]
# hddvadm -M  <Return>


Operation with non-active extension procedures   [ to procedure 1.d.]
# hddvadm -M  <Return>
already ID= 1 exists. /devices/pci@1d,2000/fibre-channel@1/sd@1,0:a,raw :
GR720
```

d. When performing pre-defined extension, the same target number cannot be registered in sd.conf and hddv.conf, so delete or comment out the conflicting definition parts in sd.conf file when definition in the two files conflict with each other.
   The following example shows the case that target=1 is repeated.

```
# vi /kernel/drv/sd.conf  <Return>
          :
name="sd" class="scsi"
     target=0 lun=0;
          :
#name="sd" class="scsi"
     target=1 lun=0;
#name="sd" class="scsi"    Delete or comment out
     target=1 lun=1;     conflicting definition parts in sd/hddv.
#name="sd" class="scsi"
     target=1 lun=2;
          :
```

   A target number that is already used by the sd driver of the system disk or CD-ROM cannot be

deleted. In this case, change the target number in hddv.conf and of the disk array device so that it does not conflict with sd.conf.

e. Edit the /kernel/drv/hddv.conf file created automatically by the hddvadm -M command, and remove comments of the LUN definition to pre-define and the LUN definition currently being used with non-active extension procedures.
If case active extension has been performed before and the LUN definition to extend the next procedures 1.f. - 1.g. are not needed.
The following example shows target=3, lun=0~2 for pre-defined extension and disk array device GR720 (target=1) in use with non-active extension procedures being registered.

```
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf
# vi /kernel/drv/hddv.conf  <Return>
          :
#name="hddv" class="scsi" target=0 lun=0;
          :
#name="hddv" class="scsi" target=0 lun=7;
          :
name="hddv" class="scsi" target=1 lun=0;
name="hddv" class="scsi" target=1 lun=1;  Remove comments of the LUN in use
name="hddv" class="scsi" target=1 lun=2;
          :
name="hddv" class="scsi" target=3 lun=0;
name="hddv" class="scsi" target=3 lun=1;  Remove comments of extended LUN
name="hddv" class="scsi" target=3 lun=2;
#name="hddv" class="scsi" target=3 lun=3;
          :
```

f. Check for LUN definition conflict between sd.conf and hddv.conf by executing the hddvchk_conf command.
The message example for "Check NG" means that target=3 is defined in both files,
and the procedures must be performed again from procedure 1.d.
The followings are examples of execution of the FJSVhddv package installation to /opt. The path of the hddvchk_conf command depends on the installed directory.

```
Check OK   [ to procedure 1.g.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
/kernel/drv/sd.conf and /kernel/drv/hddv.conf seems to be correct


Check NG   [ to procedure 1.d.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
NG ID = [ 3 ]
```

g. Reconfigure the kernel, restart the system, and then start system operation.

```
# touch /reconfigure  <Return>
# /usr/sbin/shutdown -y -i6 -g0  <Return>
```

2. Stop the machine administration hardware monitoring daemon
Execute the following commands to stop the machine administration hardware monitoring daemon.

```
# /usr/sbin/FJSVmadm/prephp  <Return>
```

3. Add PCI cards
   Add PCI cards according to the following procedure.
   To add PCI cards with the following configurations, the WWPN(World Wide Port Name: 16-digit number) of the card to add is needed.
   - If zone configuration by WWPN is done with Fibre Channel switch SN200 series.
   - If the Host Affinity function of ETERNUS3000/GR700/800 series is used.

   When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS (except for Vixel) and SP5000 SRM Facility, go to procedure 3.a, because these products get WWPNs from PCI cards automatically.
   The WWPN of the PCI card can be known from the eight characters shown on a label on the front plate of the card. These characters represent the bottom eight digits of the WWPN in hexadecimal form. The top eight digits are fixed to 10000000 in hexadecimal form.
   For example, if the following label is shown on the front plate of the card, the WWPN of the adding card is 100000000e44061.

0e24 4061

   a. Before extension of PCI cards, save the current execution result of the prtpicl(1M) command (current configuration at present.)

   ```
   # prtpicl -v > /tmp/prtpicl.pre  <Return>
   ```

   b. Check the status of the slot.
   The relationship of the position of a PCI card and format of "Ap_Id" is shown below.

Location of PCI slot	Definition of slot	Format of "Ap_Id"
Inside chassis of PRIMEPOWER 900/1500/2500	Chassis No.: **X** System board No.: **Y** Physical slot No.: **ZZ**	pcipsyN:C**X**M0**Y**-PCI#slot**ZZ**
PCI/Disk BOX	I/O Chassis No.: **X** PCI/Disk BOX No.: **Y** Physical slot No.: **ZZ**	pcipsyN:R**X**B0**Y**-PCI#slot**ZZ**
Other	Physical slot No.: **ZZ**	pcipsyN:PCI#slot**ZZ**

   Below, an example of adding a PCI card to chassis number : **0**, system board number : **1**, physical slot number : **02**,slot in chassis(C**0**M0**1**-PCI#slot**02**) is explained.

   Check the status of the target slot is "empty unconfigured."

   ```
   # cfgadm | grep C0M01-PCI#slot02  <Return>
   pcipsy15:C0M01-PCI#slot02  unknown    empty  unconfigured  unknown
   ```

   c. To confirm the slot position at extension operation, blink the ALARM LED of Ap_Id displayed in procedure 3.b.

   ```
   # cfgadm -x led=fault,mode=blink pcipsy15:C0M01-PCI#slot02  <Return>
   ```

   d. After checking that the READY LED on the PCI slot to add the PCI card is off, add PCI cards. This operation is performed by our customer support.

   e. Use the cfgadm(1M) command to check that the slot status of the added PCI card changed to "disconnected unconfigured".

   ```
   # cfgadm pcipsy15:C0M01-PCI#slot02  <Return>
   Ap_Id                       Type       Receptacle    Occupant      Condition
   pcipsy15:C0M01-PCI#slot02  unknown    disconnected  unconfigured  unknown
   ```

4. Connect PCI cards
   Connect the replaced PCI card using the cfgadm(1M) command with the configure option, or by pushing the button corresponding to the extension slot position. Note that the push button is only effective in

multiuser mode. After the new PCI card is connected, use the cfgadm(1M) command and confirm that the slot status has changed to "connected configured."
If a large-scale configuration of I/O devices is connected to the PCI card in the target slot, command execution for status confirmation may take time.

```
# cfgadm -c configure pcipsy15:C0M01-PCI#slot02  <Return>
# cfgadm pcipsy15:C0M01-PCI#slot02  <Return>
Ap_Id                       Type       Receptacle   Occupant     Condition
pcipsy15:C0M01-PCI#slot02   mult/hp    connected     configured  ok
```

5. Update hardware configuration information of machine administration/ Start the hardware monitoring daemon.
   Execute the following commands to update hardware configuration information of machine administration and to restart the hardware monitoring daemon.

```
# /usr/sbin/FJSVmadm/postphp  <Return>
```

6. Connect devices to PCI cards
   Connect I/O devices to add to the extended PCI card.
   When the Host Affinity function is being used with Fibre Channel SN200 series, Host Affinity must be configured for the device added.

   [When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS (except for Vixel) and SP5000 SRM Facility]
   WWPN zoning of SN200 series, Host affinity of Array disk device ETERNUS3000/GR700/800 series and binding setting for fjpfca.conf/sd.conf of host can be added from SAN management window automatically.
   See “Operation guide” or “User's Manual” of each product for details.
   Before using this function, the following procedure 7.a.i. and 7.a.ii. [except fcp-bind-target setting] are necessary.
   Moreover, begin from procedure 7.a.iii. after executing this function.

   [When not using SAN management function of the above products]
   SN200 series need to be reconfigured individually. For details, see "SN200 Series Affinity User's Guide" or "ETERNUS3000/GR700/800 series GRmgr User's Guide."

   Note:
   When changing Affinity configuration on SN200 series, I/O to other devices is effected by the change, and may result in temporal errors.
   I/O to disk array devices recovers normally because of retry processes, but on Fibre Channel tape devices, backup processes may end in errors. Stop backup before changing Affinity configuration.

7. Set up driver
   Add the configuration of driver according to the following operation.
   As for file related drivers, additional configuration of the PCI card driver and the device driver is needed.

   a. Add configurations of PCI card driver

      [SCSI driver(Solaris OE: glm , FUJITSU PCI SCSI Adapter Driver)]
      Configuration is not required.

      [Fibre Channel driver (FUJITSU PCI Fibre Channel)]
      i. After saving execution result of prtpicl (1M) command, take the difference between the result and the information obtained in procedure 3.a. to calculate the instance number of the fjpfca driver of the PCI card added.
         In the following example, instance number is 1, indicating that the interface name of the PCI card added is fjpfca1.
         Making the LED of the PCI card blink using the fc_adm command can specify the relation between the instance number and the slot position. The relation between the WWPN and instance number of the PCI card added can be known from the "fc_info -a" command.

```
# prtpicl -v > /tmp/prtpicl.pfca  <Return>
# diff /tmp/prtpicl.pre /tmp/prtpicl.pfca | more <Return>
```

```
          :
>   :status   okay
>   :devfs-path   /pci@8d,2000/fibre-channel@1
>   :driver-name   fjpfca
>   :binding-name  pci10cf,105c
>   :bus-addr  1
>   :instance  1
>   :_class   obp-device
>   :name  fibre-channel
          :
# /usr/sbin/FJSVpfca/fc_adm -l fjpfca1  <Return>
```

ii. Edit /kernel/drv/fjpfca.conf and add the following property definitions to the instance number from 7.a.i.

port (topology classification , link speed)
Define this property when PCI cards using Fabric connection are added.
This definition is not needed with FC-AL connection.

fcp-bind-target(connection of target ID and WWN)
Define this property when disk array equipment using Fabric connection are added.
This definition is not needed with FC-AL connection.

max-throttle(number of maximum SCSI command issue)
This property is defined for restriction on simultaneous issue of SCSI commands.
The necessity and value of this definition depends on the specification of the disk array device, so refer to the user's manual of each equipment.
The following example shows the definition of a fjpfca1 PCI card (instance number : 1) and disk array equipment(target = 3) being added.

```
# vi /kernel/drv/fjpfca.conf  <Return>
          :
port=
"fjpfca0:nport",
"fjpfca1:nport:sp1";               <- added


fcp-bind-target=
"fjpfca0t0:0x50000e1000002501",
"fjpfca1t3:0x50000e1000002ab0";   <- added


max-throttle=
"fjpfca0t0:128",
"fjpfca1t3:240";                  <- added
          :
```

iii. Execute the following command to make the fjpfca driver take in the added part.

```
# /usr/sbin/FJSVpfca/fc_config -v  <Return>
```

b. Add device driver configuration
Add device driver configuration with the following procedures according to the I/O device type.

[Disk/Tape devices(Solaris OE : sd/st)]

This example is a case where a disk device of target=2, lun=0 is pre-defined in procedure 1. and added in procedure 6. If the device is a tape device, re-read "sd" as "st" in this procedure and perform the same operation.

i. Execute the following command to make the sd driver set up the added disk device.

```
# /usr/sbin/devfsadm  <Return>
```

ii. After saving the execution result of the prtpicl(1M) command, take the difference between the result and the information obtained in 3.a. and get the physical path name of additional disk device, then get the logical path name under /dev/dsk.
In the following example, "**c2t2d0**" is the logical path name of the disk device added.

```
# prtpicl -v > /tmp/prtpicl.post  <Return>


Disk device

# diff /tmp/prtpicl.pre /tmp/prtpicl.post | grep devfs-path | grep sd
<Return>
>    :devfs-path    /pci@89,4000/scsi@2/sd@2,0
# ls -l /dev/dsk | grep /pci@89,4000/scsi@2/sd@2,0  <Return>
lrwxrwxrwx   1  root    root       41  Sep 20  22:53  c2t2d0s0
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:a
lrwxrwxrwx   1  root    root       41  Sep 20  22:53  c2t2d0s1
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:b
lrwxrwxrwx   1  root    root       41  Sep 20  22:53  c2t2d0s2
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:c
lrwxrwxrwx   1  root    root       41  Sep 20  22:53  c2t2d0s3
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:d
lrwxrwxrwx   1  root    root       41  Sep 20  22:53  c2t2d0s4
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:e
lrwxrwxrwx   1  root    root       41  Sep 20  22:53  c2t2d0s5
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:f
lrwxrwxrwx   1  root    root       41  Sep 20  22:53  c2t2d0s6
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:g
lrwxrwxrwx   1  root    root       41  Sep 20  22:53  c2t2d0s7
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:h


Tape device

# diff /tmp/prtpicl.pre /tmp/prtpicl.post | grep devfs-path | grep st
<Return>
>    :devfs-path    /pci@89,4000/scsi@2/st@5,0
# ls -l /dev/rmt | grep /pci@89,4000/scsi@2/st@5,0  <Return>
lrwxrwxrwx   1  root    root       41  Sep 20  22:53  1
-> ../../devices/pci@89,4000/scsi@2/st@5,0:
lrwxrwxrwx   1  root    root       41  Sep 20  22:53  1b
-> ../../devices/pci@89,4000/scsi@2/st@5,0:b
lrwxrwxrwx   1  root    root       41  Sep 20  22:53  1bn
-> ../../devices/pci@89,4000/scsi@2/st@5,0:bn
```

```
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1c
-> ../../devices/pci@89,4000/scsi@2/st@5,0:c
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1cb
-> ../../devices/pci@89,4000/scsi@2/st@5,0:cb
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1cbn
-> ../../devices/pci@89,4000/scsi@2/st@5,0:cbn
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1cn
-> ../../devices/pci@89,4000/scsi@2/st@5,0:cn
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1h
-> ../../devices/pci@89,4000/scsi@2/st@5,0:h
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1hb
-> ../../devices/pci@89,4000/scsi@2/st@5,0:hb
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1hbn
-> ../../devices/pci@89,4000/scsi@2/st@5,0:hbn
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1hn
-> ../../devices/pci@89,4000/scsi@2/st@5,0:hn
```

iii. If the device is a disk device, start the format(1M) command and configure labeling and partition(slice) information of the logical path name of added disks.
If the device is a tape device, this procedure is not needed.

[Disk array device (hard disk driver : hddv)]

This example is a case where a disk array device of target=3, lun=0-2 is pre-defined in procedure 1. and added in procedure 6.

i. There may be a change of status depending on the system operating state after pre-definition in procedure 1., so check the LUN definition conflict between sd.conf and hddv.conf again. In case of "Check NG", the definitions are conflicting, so go back to procedure 1.d.

```
Check OK   [ to procedure 7.b.ii.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
/kernel/drv/sd.conf and /kernel/drv/hddv.conf seems to be correct


Check NG   [ to procedure 1.d.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
NG ID = [ 3 ]
```

ii. Execute the following command to make the hard disk driver set up the LUN.

```
# /usr/sbin/hddvadmdr  <Return>
```

iii. After saving execution result of the prtpicl command, get the physical path name of added LUN by taking the difference between the result and the information obtained in 3.a, and calculate the corresponding logical path name under /dev/dsk
In the following example, "**c3t3d0**" is logical path name of added lun=0. The logical path name of lun=1,2 is set to "**c3t3d1**", "**c3t3d2**" respectively, because c**X**t**Y**d**Z**means **X**: PCI card number, **Y**: target number, **Z**: lun number .

```
# prtpicl -v > /tmp/prtpicl.post  <Return>
# diff /tmp/prtpicl.pre /tmp/prtpicl.post | grep devfs-path | grep hddv
<Return>
```

```
>    :devfs-path    /pci@89,2000/fibre-channel@1/hddv@3,0
>    :devfs-path    /pci@89,2000/fibre-channel@1/hddv@3,1
>    :devfs-path    /pci@89,2000/fibre-channel@1/hddv@3,2
# ls -l /dev/dsk | grep /pci@89,2000/fibre-channel@1/hddv@3,0  <Return>
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s0
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:a
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s1
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:b
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s2
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:c
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s3
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:d
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s4
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:e
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s5
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:f
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s6
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:g
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s7
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:h
```

iv. Start the format(1M) command and configure labeling and partition(slice) information calculated in 7.b.iii. to the logical path name of each extended LUN.

8. Setup applications
Add configuration for the application using the logical path name determined in 7.b.

[Disk device(filesystem operation)]
Create file system.
To mount at system start, add an entry in the /etc/vfstab file.

```
# newfs /dev/rdsk/c2t2d0s6  <Return>
# vi /etc/vfstab  <Return>
          :
/dev/dsk/c2t2d0s6  /dev/rdsk/c2t2d0s6   /mnt   ufs   1   yes   -
          :
```

[Disk device(raw access operation)/Tape device]
Configure applications.

For details, see the manual of each application.

[Disk device(swap device)]
Add swap device.
To make valid at system start, add an entry in the /etc/vfstab file.

```
# swap -a /dev/dsk/c2t2d0s4  <Return>
# vi /etc/vfstab  <Return>
          :
/dev/dsk/c2t2d0s4   -   -   swap    -   no  -
          :
```

```
# swap -l <Return>
swapfile               dev    swaplo  blocks  free
          :
/dev/dsk/c2t2d0s4  32,164     16  788384  683680
```

9. Start applications
   Use the logical path name determined in 7.b. to start applications.

   [Disk device(filesystem operation)]
   Mount filesystem

```
# mount -F ufs /dev/dsk/c2t2d0s6 /mnt <Return>
```

   [Disk device(raw access operation)/Tape device]
   Start applications.
   For details, see the manual of each application.

   [Disk device(swap device)]
   This procedure is not required.

3.2.1.2 Extension of PCI cards on new redundant system

The procedure to add PCI cards to a new redundant system on a Solaris 8 OE system is explained with examples for the multipath software below.

- Multipath Disk Control (MPHD)
- GR Multipath Driver (GRMPD)

If other multipath software is being used, see the manual of each product.

1. Setup devices which will be added to system
   This is an example where an disk array device at target=3, lun=0-2 is pre-defined.
   Extension using PCI Hot Plug is impossible without definition, so definition must be done before the system goes into operation.
   If this is the first time to pre-define a disk array device, begin with procedure 1.a. If this is not the first time to pre-define a disk array device, begin with procedure 1.e.

   a. Check if /kernel/drv/hddv.conf exists
      Check if /kernel/drv/hddv.conf exists. If it exists, the hard disk driver is being installed, so start from procedure 1.c. If it does not exist, the system is being operated with none-active extension procedures, so start from procedure 1.b.

```
Not exist(new install)      [ to procedure 1.c.]
# ls /kernel/drv/hddv.conf <Return>
/kernel/drv/hddv.conf: No such file or directory


Exists(operation with non-active extension procedures) [ to procedure 1.b.]
# ls /kernel/drv/hddv.conf <Return>
/kernel/drv/hddv.conf
```

   b. When operating with non-active extension procedures, execute the following commands and confirm that /kernel/drv/hddv.conf is deleted.

```
# hddvadm -i <Return>
# ls /kernel/drv/hddv.conf <Return>
/kernel/drv/hddv.conf: No such file or directory
```

   c. Use the following command to create hddv.conf file for the pre-defined extension function. The

message examples for "operation with non-active extension procedures" means that disk array device GR720 currently used with taget=1(lun=0) is registered twice on the sd/hddv driver.

```
New installation            [ to procedure 1.e.]
# hddvadm -M  <Return>


Operation with non-active extension procedures   [ to procedure 1.d.]
# hddvadm -M  <Return>
already ID= 1 exists. /devices/pci@1d,2000/fibre-channel@1/sd@1,0:a,raw :
GR720
```

d. When performing pre-defined extension, the same target number cannot be registered in sd.conf and hddv.conf, so delete or comment out the conflicting definition parts in sd.conf file when definition in the two files conflict with each other.
The following example shows the case that target=1 is repeated.

```
# vi /kernel/drv/sd.conf  <Return>
          :
name="sd" class="scsi"
     target=0 lun=0;
          :
#name="sd" class="scsi"
     target=1 lun=0;
#name="sd" class="scsi"    Delete or comment out
     target=1 lun=1;     conflicting definition parts in sd/hddv
#name="sd" class="scsi"
     target=1 lun=2;
          :
```

A target number that is already used by the sd driver of the system disk or CD-ROM cannot be deleted. In this case, change the target number in hddv.conf and of the disk array device so that it does not conflict with sd.conf.

e. Edit the /kernel/drv/hddv.conf file created automatically by the hddvadm -M command, and remove comments of the LUN definition to pre-define and the LUN definition currently being used with non-active extension procedures.
If case active extension has been performed before and the LUN definition to extend the next procedures 1.f. - 1.g. are not needed.
If the target number of each disk array controller of a redundant system differs, add definition for each target number.
The following example shows target=3 , lun=0~2 for pre-defined extension and disk array device GR720 (target=1) in use with non-active extension procedures being registered.

```
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf
# vi /kernel/drv/hddv.conf  <Return>
          :
#name="hddv" class="scsi" target=0 lun=0;
          :
#name="hddv" class="scsi" target=0 lun=7;
          :
name="hddv" class="scsi" target=1 lun=0;
```

```
name="hddv" class="scsi" target=1 lun=1;  Remove comments of the LUN in use
name="hddv" class="scsi" target=1 lun=2;
          :
name="hddv" class="scsi" target=3 lun=0;
name="hddv" class="scsi" target=3 lun=1;  Remove comments of extended LUN
name="hddv" class="scsi" target=3 lun=2;
#name="hddv" class="scsi" target=3 lun=3;
          :
```

f. Check the LUN definition overlap between sd.conf and hddv.conf by executing hddvchk_conf command.
The message example for "Check NG" means that target=3 is defined in both files,
and the procedures must be performed again from procedure 1.d.
The followings are examples of execution of the FJSVhddv package installation to /opt. The path of the hddvchk_conf command depends on the installed directory.

```
Check OK   [ to procedure 1.g.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
/kernel/drv/sd.conf and /kernel/drv/hddv.conf seems to be correct


Check NG   [ to procedure 1.d.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
NG ID = [ 3 ]
```

g. Reconfigure the kernel , restart the system, and then start system operation.

```
# touch /reconfigure  <Return>
# /usr/sbin/shutdown -y -i6 -g0  <Return>
```

2. Stop the machine administration monitoring daemon
Use the following command to stop the hardware disorder monitoring daemon of machine administration.

```
# /usr/sbin/FJSVmadm/prephp   <Return>
```

3. Add PCI cards
Add PCI cards according to the following procedure.
To add PCI cards with the following configurations, the WWPN(World Wide Port Name: 16-digit number) of the card to add is needed.
   - If zone configuration by WWPN is done with Fibre Channel switch SN200 series.
   - If the Host Affinity function of ETERNUS3000/GR700/800 series is used.

When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS (except for Vixel) and SP5000 SRM Facility, go to procedure 3.a, because these products get WWPNs from PCI cards automatically.
The WWPN of the PCI card can be known from the eight characters shown on a label on the front plate of the card. These characters represent the bottom eight digits of the WWPN in hexadecimal form. The top eight digits are fixed to 10000000 in hexadecimal form.
For example, if the following label is shown on the front plate of the card, the WWPN of the adding card is 100000000e44061.

```
0e24
4061
```

a. Before extension of PCI cards, save the current execution result of the prtpicl(1M) command (current configuration at present.)

```
# prtpicl -v > /tmp/prtpicl.pre  <Return>
```

b. Check the status of the slot.
The relationship of the position of a PCI card and format of "Ap_Id" is shown below.

Location of PCI slot	Definition of slot	Format of "Ap_Id"
Inside chassis of PRIMEPOWER 900/1500/2500	Chassis No.: **X** System board No.: **Y** Physical slot No.: **ZZ**	pcipsyN:C**X**M0**Y**-PCI#slot**ZZ**
PCI/Disk BOX	I/O Chassis No.: **X** PCI/Disk BOX No.: **Y** Physical slot No.: **ZZ**	pcipsyN:R**X**B0**Y**-PCI#slot**ZZ**
Other	Physical slot No.: **ZZ**	pcipsyN:PCI#slot**ZZ**

Below, an example of adding a PCI card to chassis number : **0**, PCI/Disk BOX number : **1**, physical slot number : **02**, slot in PCI/Disk BOX(R**0**B0**1**-PCI#slot**02**) is explained.

Check the status of the target slot is "empty unconfigured."

```
# cfgadm | grep R0B01-PCI#slot02  <Return>
pcipsy18:R0B01-PCI#slot02  unknown    empty  unconfigured  unknown
```

c. To confirm the slot position at extension operation, blink the ALARM LED of Ap_Id displayed in procedure 3.b.

```
# cfgadm -x led=fault,mode=blink pcipsy18:R0B01-PCI#slot02  <Return>
```

d. After checking that the READY LED on the PCI slot to add the PCI card is off, add PCI cards. This operation is performed by our customer support.

e. Use the cfgadm(1M) command to check that the slot status of the added PCI card changed to "disconnected unconfigured".

```
# cfgadm pcipsy18:R0B01-PCI#slot02  <Return>
Ap_Id                      Type       Receptacle    Occupant      Condition
pcipsy18:R0B01-PCI#slot02  unknown    disconnected  unconfigured  unknown
```

4. Connect the PCI card
Connect the replaced PCI card using the cfgadm(1M) command with the configure option, or by pushing the button corresponding to the extension slot position. Note that the push button is only effective in multiuser mode. After the new PCI card is connected, use the cfgadm(1M) command and confirm that the slot status has changed to "connected configured."
If a large-scale configuration of I/O devices is connected to the PCI card in the target slot, command execution for status confirmation may take time.

```
# cfgadm -c configure pcipsy18:R0B01-PCI#slot02  <Return>
# cfgadm pcipsy18:R0B01-PCI#slot02  <Return>
Ap_Id                      Type       Receptacle   Occupant     Condition
pcipsy18:R0B01-PCI#slot02  mult/hp    connected    configured   ok
```

5. Update hardware configuration information of machine administration/ Start the hardware monitoring daemon.
Execute the following commands to update hardware configuration information of machine administration and to restart the hardware monitoring daemon..

```
# /usr/sbin/FJSVmadm/postphp  <Return>
```

6. Connect devices to PCI cards
Connect new disk array device in redundant configuration to the extended PCI card.
When the Host Affinity function is being used with Fibre Channel switch SN200 series, Host Affinity must be configured for the device added.

[When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS

(except for Vixel) and SP5000 SRM Facility]
WWPN zoning of SN200 series, Host affinity of Array disk device ETERNUS3000/GR700/800 series and binding setting for fjpfca.conf/sd.conf of host can be added from SAN management window automatically.
See “Operation guide” or “User's Manual” of each product for details.
Before using this function, the following procedure 7.a.i. and 7.a.ii. [except fcp-bind-target setting] are necessary.
Moreover, begin from procedure 7.a.iii. after executing this function.

[When not using SAN management function of the above products]
SN200 series need to be reconfigured individually. For details, see "SN200 Series Affinity User's Guide" or "ETERNUS3000/GR700/800 series GRmgr User's Guide."

Note:
When changing Affinity configuration on SN200 series, I/O to other devices is effected by the change, and may result in temporal errors.
I/O to disk array devices recovers normally because of retry processes, but on Fibre Channel tape devices, backup processes may end in errors. Stop backup before changing Affinity configuration.

7. Set up driver
Add the configuration of driver according to the following operation.
As for file related drivers, additional configuration of the PCI card driver and the device driver is needed.

   a. Add configuration for PCI card driver

[SCSI driver(Solaris OE:glm, FUJITSU PCI SCSI Adapter Driver)]
Configuration is not required.

[Fibre Channel driver(FUJITSU PCI Fibre Channel)]

   i. After saving execution result of prtpicl (1M) command, take the difference between the result and the information obtained in procedure 3.a. to calculate the instance number of the fjpfca driver of the PCI card added.
   In the following example, instance number is 1, indicating that the interface name of the PCI card added is fjpfca1.
   Making the LED of the PCI card blink using the fc_adm command can specify the relation between the instance number and the slot position. The relation between the WWPN and instance number of the PCI card added can be known from the "fc_info -a" command.

```
# prtpicl -v > /tmp/prtpicl.pfca  <Return>
# diff /tmp/prtpicl.pre /tmp/prtpicl.pfca | more <Return>
          :
>   :status   okay
>   :fru   PCI slot(PCI#02 at Rack#0,IO_BOX#1)
>   :component-name   R0B01-PCI#02
>   :devfs-path   /pci@8d,2000/fibre-channel@1
>   :driver-name   fjpfca
>   :binding-name  pci10cf,105c
>   :bus-addr  1
>   :instance  1
>   :_class   obp-device
>   :name  fibre-channel
          :
```

   ii. Edit the /kernel/drv/fjpfca.conf file, and add the following property definitions to the instance number determined in 7.a.i.

port(type of topology, link speed)
Defined when adding PCI cards with Fabric connection.
This definition is not needed with FC-AL connection.

fcp-bind-target(connection between target ID and WWN)
Defined when adding disk array with Fabric connection.
This definition is not needed with FC-AL connection.

max-throttle(maximum issued SCSI commands)
Defined to limit the number of SCSI commands issued at the same time.
The need and value of this definition depends on the disk array device, so refer to the manuals of each disk array device. Below is an example where definitions for a PCI card at fjpfca1(instance number 1) and a disk array device at target=3 is added.

```
# vi /kernel/drv/fjpfca.conf  <Return>
          :
port=
"fjpfca0:nport",
"fjpfca1:nport:sp1";               <- add

fcp-bind-target=
"fjpfca0t0:0x50000e1000002501",
"fjpfca1t3:0x50000e1000002ab0";    <- add

max-throttle=
"fjpfca0t0:128",
"fjpfca1t3:240";                  <- add
          :
```

iii. Execute the following command to make the fjpfca driver set up the additional part.

```
# /usr/sbin/FJSVpfca/fc_config -v  <Return>
```

b. Add configuration of device driver

An example where a path with target=3, lun=0-2 is added to a disk array device with procedure 6. is shown.

i. The status may have changed after the pre-definition in 1., so check again the LUN definition overlap between sd.conf and hddv.conf by executing hddvchk_conf command. "Check NG" means that definitions are repeated in the two files, so go back to 1.d.

```
Check OK   [ to procedure 7.b.ii.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
/kernel/drv/sd.conf and /kernel/drv/hddv.conf seems to be correct


Check NG   [ to procedure 1.d.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
NG ID = [ 3 ]
```

ii. Execute the following command in order to indicate connecting extension LUN to hard disk driver.

```
# /usr/sbin/hddvadmdr  <Return>
```

iii. After saving executing result of prtpicl command, get the physical path name of added LUN by taking difference between that result and extracted information of 3.a, and calculate the corresponding logical path name under /dev/dsk

The following examples shows that "**c3t3d0**" is logical path name of extended lun=0. The logical path name of lun=1,2 is set to "**c3t3d1**", "**c3t3d2**" respectively, because c**X**t**Y**d**Z**show **X**: PCI card number, **Y**: target number, **Z**: lun number .

```
# prtpicl -v > /tmp/prtpicl.post  <Return>
# diff /tmp/prtpicl.pre /tmp/prtpicl.post | grep devfs-path | grep hddv
<Return>
>    :devfs-path    /pci@8d,2000/fibre-channel@1/hddv@3,0
>    :devfs-path    /pci@8d,2000/fibre-channel@1/hddv@3,1
>    :devfs-path    /pci@8d,2000/fibre-channel@1/hddv@3,2
>    :devfs-path    /pci@89,2000/fibre-channel@1/hddv@3,0
>    :devfs-path    /pci@89,2000/fibre-channel@1/hddv@3,1
>    :devfs-path    /pci@89,2000/fibre-channel@1/hddv@3,2
# ls -l /dev/dsk | grep /pci@8d,2000/fibre-channel@1/hddv@3,0  <Return>
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s0
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:a
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s1
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:b
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s2
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:c
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s3
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:d
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s4
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:e
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s5
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:f
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s6
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:g
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s7
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:h
# ls -l /dev/dsk | grep /pci@89,2000/fibre-channel@1/hddv@3,0  <Return>
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c4t3d0s0
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:a
             :
```

iv. Start formats (1M) command and configure labeling and partition (slice) information calculated at 6.b.iii to the logical path name of each extended LUN. This operation is only needed for each LUN on the representative additional path, and is not needed to be performed on each additional PCI card.

8. Setup redundant system
   Execute the following commands to create a new redundant system.
   In case of MPHD, the commands need to be executed for each LUN, specifying the path name determined in 7.b.iii as a parameter.
   The italic letters below are the new multipath names, and the underlined letters are the path names that compose the multipath.

```
MPHD
# /usr/opt/FJSViomp/bin/iompadm -c mphd new /dev/rdsk/c3t3d0s2 /dev/rdsk/c4t3d0s2
<Return>
```

```
IOMP: /dev/FJSVmphd/fiomp/adm2
Node:
       /dev/FJSVmphd/rdsk/mphd2s0
       /dev/FJSVmphd/rdsk/mphd2s1
       /dev/FJSVmphd/rdsk/mphd2s2
       /dev/FJSVmphd/rdsk/mphd2s3
       /dev/FJSVmphd/rdsk/mphd2s4
       /dev/FJSVmphd/rdsk/mphd2s5
       /dev/FJSVmphd/rdsk/mphd2s6
       /dev/FJSVmphd/rdsk/mphd2s7

GRMPD
# /usr/sbin/mplbconfig -nv  <Return>
*** Phase 1: checking /dev/rdsk ***
*** Phase 2: checking multi path ***
*** Phase 3: analyzing path ***
     Path    : Action : Element path          : LUN : Storage        : Zone
     mplb0   : exist  : c1t4d0s2 c2t3d0s2      : 0 : GR720464f-12340001 :
       :
     mplb8   : new    : c3t3d0s2 c4t3d0s2      : 0 : GR8404550-  000213 :
       :
# /usr/sbin/mplbconfig -av  <Return>
              :
IOMP: /dev/FJSVmplb/fiomp/adm8
Node:
       /dev/FJSVmplb/rdsk/mplb8s0
       /dev/FJSVmplb/rdsk/mplb8s1
       /dev/FJSVmplb/rdsk/mplb8s2
       /dev/FJSVmplb/rdsk/mplb8s3
       /dev/FJSVmplb/rdsk/mplb8s4
       /dev/FJSVmplb/rdsk/mplb8s5
       /dev/FJSVmplb/rdsk/mplb8s6
       /dev/FJSVmplb/rdsk/mplb8s7
              :
```

9. Setup applications
   Add configuration for the application using the logical path name determined in 8.

   [filesystem operation]
   Create file system.
   To mount at system start, add an entry in the /etc/vfstab file.

```
# newfs /dev/FJSVmphd/rdsk/mphd2s6  <Return>
# vi /etc/vfstab  <Return>
          :
/dev/FJSVmphd/dsk/mphd2s6  /dev/FJSVmphd/rdsk/mphd2s6   /mnt   ufs   1   yes   -
```

```
          :
```

[raw access operation]
Configure applications.
For details, see the manual of each application.

[swap device]
Add swap device.
To make valid at system start, add an entry in the /etc/vfstab file.

```
# swap -a /dev/FJSVmphd/dsk/mphd2s4  <Return>
# vi /etc/vfstab  <Return>
          :
/dev/FJSVmphd/dsk/mphd2s4   -   -   swap    -   no  -
          :
# swap -l  <Return>
swapfile               dev    swaplo  blocks  free
          :
/dev/FJSVmphd/dsk/mphd2s4  32,164     16  788384  683680
```

10. Start applications
    Use the logical path name determined in 8. to start applications.

    [filesystem operation]
    Mount filesystem

```
# mount -F ufs /dev/FJSVmphd/dsk/mphd2s6 /mnt  <Return>
```

    [raw access operation]
    Start applications.
    For details, see the manual of each application.

    [swap device]
    This procedure is not required.

3.2.1.3 Extension of PCI cards to redundant system

The procedure to add PCI cards to a redundant system in order to extend redundancy on a Solaris 8 OE system is explained with an example with GR multipath driver (GRMPD).

There are no additional patterns for Multipath Disk Control (MPHD).
If multipath software products other than GRMPD are being used, see the manual of each product.

No configuration is needed for applications when adding PCI cards to an existing redundant system.

1. Setup devices which will be added to system
   This is an example where a disk array controller of target=3, lun=0 is added with pre-definition to an existing disk array device. Extension using PCI Hot Plug cannot be done without definition, so devices must be defined before the system goes into operation. If this is the first time to pre-define a disk array device, begin with procedure 1.a. If this is not the first time to pre-define a disk array device, begin with procedure 1.e.

   a. Check if /kernel/drv/hddv.conf exists
      Check if /kernel/drv/hddv.conf exists. If it does not exist, the hard disk driver is being installed, so start from procedure 1.c. If it does exist, the system is being operated with non-active extension procedures, so start from procedure 1.b.

```
Not exist(new install)      [ to procedure 1.c.]
# ls /kernel/drv/hddv.conf  <Return>
```

```
/kernel/drv/hddv.conf: No such file or directory


Exists(operation with non-active extension procedures) [ to procedure 1.b.]
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf
```

b. When operating with non-active extension procedures, execute the following commands and confirm that /kernel/drv/hddv.conf is deleted

```
# hddvadm -i  <Return>
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf: No such file or directory
```

c. Use the following command to create an hddv.conf file for the pre-defined extension function.
The message examples for "operation with non-active extension procedures" means that GR720 currently used with taget=1(lun=0) is registered twice on the sd/hddv driver.

```
New installation            [ to procedure 1.e.]
# hddvadm -M  <Return>


Set up with a non-active extension procedure   [ to procedure 1.d.]
# hddvadm -M  <Return>
already ID= 1 exists. /devices/pci@1d,2000/fibre-channel@1/sd@1,0:a,raw : GR720
```

d. When performing pre-defined extension, the same target number cannot be registered in sd.conf and hddv.conf, so delete or comment out the conflicting definition parts in sd.conf file when definition in the two files conflict with each other.
The following example shows the case that target=1 is repeated.

```
# vi /kernel/drv/sd.conf  <Return>
          :
name="sd" class="scsi"
     target=0 lun=0;
          :
#name="sd" class="scsi"
     target=1 lun=0;
#name="sd" class="scsi"    Delete or comment out
     target=1 lun=1;     the duplication definition parts in sd/hddv
#name="sd" class="scsi"
     target=1 lun=2;
          :
```

A target number that is already used by the sd driver of the system disk or CD-ROM cannot be deleted. In this case, change the target number in hddv.conf and of the disk array device so that it does not conflict with sd.conf.

e. Edit the /kernel/drv/hddv.conf file created automatically by the hddvadm -M command, and remove comments of the LUN definition to pre-define and the LUN definition currently being used with non-active extension procedures.
If case active extension has been performed before and the LUN definition to extend the next procedures 1.f. - 1.g. are not needed.
The following example shows target=3 , lun=0~2 for pre-defined extension and disk array device

GR720 (target=1) in use with non-active extension procedures being registered.

```
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf
# vi /kernel/drv/hddv.conf  <Return>
          :
#name="hddv" class="scsi" target=0 lun=0;
          :
#name="hddv" class="scsi" target=0 lun=7;
          :
name="hddv" class="scsi" target=1 lun=0;
name="hddv" class="scsi" target=1 lun=1;  remove comments of the LUN in use
name="hddv" class="scsi" target=1 lun=2;
          :
name="hddv" class="scsi" target=3 lun=0;
name="hddv" class="scsi" target=3 lun=1;  remove comments of extended LUN
name="hddv" class="scsi" target=3 lun=2;
#name="hddv" class="scsi" target=3 lun=3;
          :
```

f. Check for LUN definition conflict between sd.conf and hddv.conf by executing the hddvchk_conf command.
The message example for "Check NG" means that target=3 is defined in both files,
and the procedures must be performed again from procedure 1.d.
The followings are examples of execution of the FJSVhddv package installation to /opt. The path of the hddvchk_conf command depends on the installed directory.

```
Check OK   [ to procedure 1.g.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
/kernel/drv/sd.conf and /kernel/drv/hddv.conf seems to be correct


Check NG   [ to procedure 1.d.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
NG ID = [ 3 ]
```

g. Reconfigure the kernel , restart the system, and then start system operation.

```
# touch /reconfigure  <Return>
# /usr/sbin/shutdown -y -i6 -g0  <Return>
```

2. Stop the machine administration hardware monitoring daemon
Execute the following commands to stop the machine administration hardware monitoring daemon.

```
# /usr/sbin/FJSVmadm/prephp  <Return>
```

3. Add PCI cards
Add PCI cards according to the following procedure.
To add PCI cards with the following configurations, the WWPN(World Wide Port Name: 16-digit number) of the card to add is needed.
   - If zone configuration by WWPN is done with Fibre Channel switch SN200 series.
   - If the Host Affinity function of ETERNUS3000/GR700/800 series is used.

When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS (except for Vixel) and SP5000 SRM Facility, go to procedure 3.a, because these products get WWPNs from

PCI cards automatically.
The WWPN of the PCI card can be known from the eight characters shown on a label on the front plate of the card. These characters represent the bottom eight digits of the WWPN in hexadecimal form. The top eight digits are fixed to 10000000 in hexadecimal form.
For example, if the following label is shown on the front plate of the card, the WWPN of the adding card is 100000000e44061.

0e24 4061

a. Before extension of PCI cards, save the current execution result of the prtpicl(1M) command (current configuration at present.)

```
# prtpicl -v > /tmp/prtpicl.pre <Return>
```

b. Check the status of the slot.
When adding PCI cards to a redundant configuration, several PCI cards may be added to the system. In this case, perform to procedures 4. on each card.
The relationship of the position of a PCI card and format of "Ap_Id" is shown below.

Location of PCI slot	Definition of slot	Format of "Ap_Id"
Inside chassis of PRIMEPOWER 900/1500/2500	Chassis No.: **X** System board No.: **Y** Physical slot No.: **ZZ**	pcipsyN:C**X**M0**Y**-PCI#slot**ZZ**
PCI/Disk BOX	I/O Chassis No.: **X** PCI/Disk BOX No.: **Y** Physical slot No.: **ZZ**	pcipsyN:R**X**B0**Y**-PCI#slot**ZZ**
Other	Physical slot No.: **ZZ**	pcipsyN:PCI#slot**ZZ**

Below, an example of adding a PCI card to chassis number : **0**, PCI/Disk BOX number : **1**, physical slot number : **02**, slot in PCI/Disk BOX(R**0**B0**1**-PCI#slot**02**) is explained.

Check the status of the target slot is "empty unconfigured."

```
# cfgadm | grep R0B01-PCI#slot02 <Return>
pcipsy18:R0B01-PCI#slot02  unknown    empty  unconfigured  unknown
```

c. To confirm the slot position at extension operation, blink the ALARM LED of Ap_Id displayed in procedure 3.b.

```
# cfgadm -x led=fault,mode=blink pcipsy18:R0B01-PCI#slot02 <Return>
```

d. After checking that the READY LED on the PCI slot to add the PCI card is off, add PCI cards. This operation is performed by our customer support.

e. Use the cfgadm(1M) command to check that the slot status of the added PCI card changed to "disconnected unconfigured".

```
# cfgadm pcipsy18:R0B01-PCI#slot02 <Return>
Ap_Id                       Type       Receptacle    Occupant      Condition
pcipsy18:R0B01-PCI#slot02  unknown    disconnected  unconfigured  unknown
```

4. Connect PCI cards
Connect the replaced PCI card using the cfgadm(1M) command with the configure option, or by pushing the button corresponding to the extension slot position. Note that the push button is only effective in multi-user mode. After the new PCI card is connected, use the cfgadm(1M) command and confirm that the slot status has changed to "connected configured."
If a large-scale configuration of I/O devices is connected to the PCI card in the target slot, command execution for status confirmation may take time.

```
# cfgadm -c configure pcipsy18:R0B01-PCI#slot02 <Return>
# cfgadm pcipsy18:R0B01-PCI#slot02 <Return>
```

```
Ap_Id                        Type       Receptacle   Occupant     Condition
pcipsy18:R0B01-PCI#slot02    mult/hp     connected    configured    ok
```

5. Update hardware configuration information of machine administration/ Start the hardware monitoring daemon.
   Execute the following commands to update hardware configuration information of machine administration and to restart the hardware monitoring daemon.

```
# /usr/sbin/FJSVmadm/postphp  <Return>
```

6. Connect equipment to PCI cards
   Add controllers to an existing disk array device, and connect to the PCI card added.
   When the Host Affinity function is being used with Fibre Channel SN200 series, Host Affinity must be configured for the device added.

   [When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS (except for Vixel) and SP5000 SRM Facility]
   WWPN zoning of SN200 series, Host affinity of Array disk device ETERNUS3000/GR700/800 series and binding setting for fjpfca.conf/sd.conf of host can be added from SAN management window automatically.
   See “Operation guide” or “User's Manual” of each product for details.
   Before using this function, the following procedure 7.a.i. and 7.a.ii. [except fcp-bind-target setting] are necessary.
   Moreover, begin from procedure 7.a.iii. after executing this function.

   [When not using SAN management function of the above products]
   SN200 series need to be reconfigured individually. For details, see "SN200 Series Affinity User's Guide" or "ETERNUS3000/GR700/800 series GRmgr User's Guide."

   Note:
   When changing Affinity configuration on SN200 series, I/O to other devices is effected by the change, and may result in temporal errors.
   I/O to disk array devices recovers normally because of retry processes, but on Fibre Channel tape devices, backup processes may end in errors. Stop backup before changing Affinity configuration.

7. Set up driver
   Add the configuration of driver according to the following operation.
   As to file related driver, additional configuration of PCI cards driver and device driver is needed.

   a. Add the configuration of PCI cards driver

      [SCSI driver(Solaris OE: glm , FUJITSU PCI SCSI Adapter Driver)]
      Configuration is not required.

      [Fibre Channel driver (FUJITSU PCI Fibre Channel)]

      i. After saving execution result of prtpicl (1M) command, take the difference between the result and the information obtained in procedure 3.a. to calculate the instance number of the fjpfca driver of the PCI card added.
         In the following example, instance number is 1, indicating that the interface name of the PCI card added is fjpfca1.
         Making the LED of the PCI card blink using the fc_adm command can specify the relation between the instance number and the slot position. The relation between the WWPN and instance number of the PCI card added can be known from the "fc_info -a" command

```
# prtpicl -v > /tmp/prtpicl.pfca  <Return>
# diff /tmp/prtpicl.pre /tmp/prtpicl.pfca | more <Return>
          :
>   :status   okay
>   :devfs-path   /pci@8d,2000/fibre-channel@1
>   :driver-name   fjpfca
>   :binding-name  pci10cf,105c
```

```
>    :bus-addr  1
>    :instance  1
>    :_class    obp-device
>    :name  fibre-channel
           :
# /usr/sbin/FJSVpfca/fc_adm -l fjpfca1  <Return>
```

ii. Edit /kernel/drv/fjpfca.conf definitions to the instance number from 7.a.i. and add the following property

port (topology classification , link speed)
Define this property when PCI cards using Fabric connection are added.
This definition is not needed with FC-AL connection.

fcp-bind-target(connection of target ID and WWN)
Define this property when disk array equipment using Fabric connection is added.
This definition is not needed with FC-AL connection.

max-throttle(number of maximum SCSI command issue)
This property is defined for restriction on simultaneous issue of SCSI commands.
The necessity and value of this definition depends on the specification of the disk array device, so refer to the user's manual of each equipment.
The following example shows the definition of a fjpfca1 PCI card (instance number : 1) and disk array equipment(target = 3) being added.

```
# vi /kernel/drv/fjpfca.conf  <Return>
           :
port=
"fjpfca0:nport",
"fjpfca1:nport:sp1";                <- added

fcp-bind-target=
"fjpfca0t0:0x50000e1000002501",
"fjpfca1t3:0x50000e1000002ab0";    <- added

max-throttle=
"fjpfca0t0:128",
"fjpfca1t3:240";                   <- added
           :
```

iii. Execute the following command to make the fjpfca driver take in the added part.

```
# /usr/sbin/FJSVpfca/fc_config -v  <Return>
```

b. Add device driver configuration
An example where a path with target=3, lun=0-2 is added to a disk array device with procedure 6 is shown.

i. The status may have changed after the pre-definition in 1., so check again the LUN definition overlap between sd.conf and hddv.conf by executing hddvchk_conf command.
"Check NG" means that definitions are repeated in the two files, so go back to 1.d.

```
Check OK    [ to procedure 7.b.ii.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
```

```
/kernel/drv/sd.conf and /kernel/drv/hddv.conf seems to be correct


Check NG   [ to procedure 1.d.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
NG ID = [ 3 ]
```

ii. Execute the following command in order to indicate connecting extension LUN to hard disk driver.

```
# /usr/sbin/hddvadmdr  <Return>
```

iii. After saving executing result of prtpicl command, get the physical path name of added LUN by taking difference between that result and extracted information of 3.a, and calculate the corresponding logical path name under /dev/dsk
The following examples shows that "**c3t3d0**" is logical path name of extended lun=0. The logical path name of lun=1,2 is set to "**c3t3d1**", "**c3t3d2**" respectively, because c**X**t**Y**d**Z**show **X**: PCI card number, **Y**: target number, **Z**: lun number .

```
# prtpicl -v > /tmp/prtpicl.post  <Return>
# diff /tmp/prtpicl.pre /tmp/prtpicl.post | grep devfs-path | grep hddv
<Return>
>    :devfs-path    /pci@8d,2000/fibre-channel@1/hddv@3,0
>    :devfs-path    /pci@8d,2000/fibre-channel@1/hddv@3,1
>    :devfs-path    /pci@8d,2000/fibre-channel@1/hddv@3,2
# ls -l /dev/dsk | grep /pci@8d,2000/fibre-channel@1/hddv@3,0  <Return>
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s0
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:a
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s1
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:b
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s2
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:c
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s3
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:d
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s4
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:e
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s5
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:f
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s6
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:g
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s7
-> ../../devices/pci@8d,2000/fibre-channel@1/hddv@3,0:h
```

8. Connect to redundant system
To connect the added PCI card to the redundant system, execute the following command and confirm that the path name determined in procedure 7.b.iii. (in underlined letters) is added.

```
# /usr/sbin/mplbconfig -nv  <Return>
*** Phase 1: checking /dev/rdsk ***
*** Phase 2: checking multi path ***
*** Phase 3: analyzing path ***
```

```
     Path    : Action : Element path          : LUN : Storage        : Zone
     mplb0   : exist  : c1t4d0s2 c2t3d0s2      : 0 : GR720464f-12340001 :
       :
     mplb8   : add    : c3t3d0s2              : 0 : GR8404550-  000213 :
       :
# /usr/sbin/mplbconfig -av  <Return>
             :
IOMP: /dev/FJSVmplb/fiomp/adm8
Node:
       /dev/FJSVmplb/rdsk/mplb8s0
       /dev/FJSVmplb/rdsk/mplb8s1
       /dev/FJSVmplb/rdsk/mplb8s2
       /dev/FJSVmplb/rdsk/mplb8s3
       /dev/FJSVmplb/rdsk/mplb8s4

       /dev/FJSVmplb/rdsk/mplb8s5

       /dev/FJSVmplb/rdsk/mplb8s6

       /dev/FJSVmplb/rdsk/mplb8s7

             :
```

3.2.2 Solaris 9 OE

This section describes the procedures to extend PCI cards on Solaris 9 OE systems.

3.2.2.1 Extension of PCI cards on non-redundant system

Procedures to add PCI cards without making them redundant on a Solaris 8 OE system are explained here.

1. Stop the machine administration hardware monitoring daemon
   Execute the following commands to stop the machine administration hardware error monitoring daemon.

```
# /usr/sbin/FJSVmadm/prephp  <Return>
```

2. Add PCI cards
   Add PCI cards according to the following procedure.
   To add PCI cards with the following configurations, the WWPN(World Wide Port Name: 16-digit number) of the card to add is needed.
   - If zone configuration by WWPN is done with Fibre Channel switch SN200 series.
   - If the Host Affinity function of ETERNUS3000/GR700/800 series is used.

   When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS (except for Vixel) and SP5000 SRM Facility, go to procedure 2.a, because these products get WWPNs from PCI cards automatically.
   The WWPN of the PCI card can be known from the eight characters shown on a label on the front plate of the card. These characters represent the bottom eight digits of the WWPN in hexadecimal form. The top eight digits are fixed to 10000000 in hexadecimal form.
   For example, if the following label is shown on the front plate of the card, the WWPN of the adding card is 100000000e44061.

```
0e24
4061
```

a. Before extension of PCI cards, save the current execution result of prtpicl(1M) command (composition information at present)

```
# prtpicl -v > /tmp/prtpicl.pre  <Return>
```

b. Check the status of the slot.
The relationship of the position of a PCI card and format of "Ap_Id" is shown below.

Location of PCI slot	Definition of slot	Format of "Ap_Id"
Inside chassis of PRIMEPOWER 900/1500/2500	Chassis No.: **X** System board No.: **Y** Physical slot No.: **ZZ**	pcipsyN:C**X**M0**Y**-PCI#slot**ZZ**
PCI/Disk BOX	I/O Chassis No.: **X** PCI/Disk BOX No.: **Y** Physical slot No.: **ZZ**	pcipsyN:R**X**B0**Y**-PCI#slot**ZZ**
Other	Physical slot No.: **ZZ**	pcipsyN:PCI#slot**ZZ**

Below, an example of adding a PCI card to chassis number : **0**, system board number : **1**, physical slot number : **02**,slot in chassis(C**0**M0**1**-PCI#slot**02**) is explained.

Check the status of the target slot is "empty unconfigured."

```
# cfgadm | grep C0M01-PCI#slot02  <Return>
pcipsy15:C0M01-PCI#slot02  unknown    empty  unconfigured  unknown
```

c. To confirm the slot position at extension operation, blink the ALARM LED of Ap_Id displayed in procedure 2.b.

```
# cfgadm -x led=fault,mode=blink pcipsy15:C0M01-PCI#slot02  <Return>
```

d. After checking that the READY LED on the PCI slot to add the PCI card is off, add PCI cards. This operation is performed by our customer support.

e. Use the cfgadm(1M) command to check that the slot status of the added PCI card changed to "disconnected unconfigured".

```
# cfgadm pcipsy15:C0M01-PCI#slot02  <Return>
Ap_Id                       Type       Receptacle    Occupant      Condition
pcipsy15:C0M01-PCI#slot02  unknown    disconnected  unconfigured  unknown
```

3. Connect PCI cards
Connect the replaced PCI card using the cfgadm(1M) command with the configure option, or by pushing the button corresponding to the extension slot position. Note that the push button is only effective in multiuser mode. After the new PCI card is connected, use the cfgadm(1M) command and confirm that the slot status has changed to "connected configured."
If a large-scale configuration of I/O devices is connected to the PCI card in the target slot, command execution for status confirmation may take time.

```
# cfgadm -c configure pcipsy15:C0M01-PCI#slot02  <Return>
# cfgadm pcipsy15:C0M01-PCI#slot02  <Return>
Ap_Id                        Type        Receptacle  Occupant       Condition
pcipsy15:C0M01-PCI#slot02    mult/hp     connected   configured     ok
```

4. Update hardware configuration information of machine administration/ Start the hardware monitoring daemon.
Execute the following commands to update hardware configuration information of machine administration and to restart the hardware monitoring daemon.

```
# /usr/sbin/FJSVmadm/postphp  <Return>
```

5. Connect equipment to PCI cards
   Connect I/O devices to add to the extended PCI card.
   When the Host Affinity function is being used with Fibre Channel SN200 series, Host Affinity must be configured for the device added.

   [When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS (except for Vixel) and SP5000 SRM Facility]
   WWPN zoning of SN200 series, Host affinity of Array disk device ETERNUS3000/GR700/800 series and binding setting for fjpfca.conf/sd.conf of host can be added from SAN management window automatically.
   See “Operation guide” or “User's Manual” of each product for details.
   Before using this function, the following procedure 6.a.i. and 6.a.ii. [except fcp-bind-target setting] are necessary.
   Moreover, begin from procedure 6.a.iii. after executing this function.

   [When not using SAN management function of the above products]
   SN200 series need to be reconfigured individually. For details, see "SN200 Series Affinity User's Guide" or "ETERNUS3000/GR700/800 series GRmgr User's Guide."

   Note:
   When changing Affinity configuration on SN200 series, I/O to other devices is effected by the change, and may result in temporal errors.
   I/O to disk array devices recovers normally because of retry processes, but on Fibre Channel tape devices, backup processes may end in errors. Stop backup before changing Affinity configuration.

6. Set up driver
   Add the configuration of driver according to the following operation.
   As for file related drivers, additional configuration of the PCI card driver and the device driver is needed.

   a. Add the configuration of PCI cards driver

      [SCSI driver (Solaris OE: glm , FUJITSU PCI SCSI Adapter Driver)]
      Configuration is not required.

      [Fibre Channel driver (FUJITSU PCI Fibre Channel)]
      i. After saving execution result of prtpicl (1M) command, take the difference between the result and the information obtained in procedure 2.a. to calculate the instance number of the fjpfca driver of the PCI card added.
         In the following example, instance number is 1, indicating that the interface name of the PCI card added is fjpfca1.
         Making the LED of the PCI card blink using the fc_adm command can specify the relation between the instance number and the slot position. The relation between the WWPN and instance number of the PCI card added can be known from the "fc_info -a" command

```
# prtpicl -v > /tmp/prtpicl.pfca  <Return>
# diff /tmp/prtpicl.pre /tmp/prtpicl.pfca | more <Return>
          :
>    :status    okay
>    :devfs-path    /pci@8d,2000/fibre-channel@1
>    :driver-name    fjpfca
>    :binding-name  pci10cf,105c
>    :bus-addr  1
>    :instance  1
>    :_class    obp-device
>    :name  fibre-channel
          :
# /usr/sbin/FJSVpfca/fc_adm -l fjpfca1  <Return>
```

      ii. Edit /kernel/drv/fjpfca.conf and add the following property definitions to the instance

number from 6.a.i.

port (topology classification , link speed)
Define this property when PCI cards using Fabric connection are added.
This definition is not needed with FC-AL connection.

fcp-bind-target(connection of target ID and WWN)
Define this property when disk array equipment using Fabric connection are added.
This definition is not needed with FC-AL connection.

max-throttle(number of maximum SCSI command issue)
This property is defined for restriction on simultaneous issue of SCSI commands.
The necessity and value of this definition depends on the specification of the disk array device, so refer to the user's manual of each equipment.
The following example shows the definition of a fjpfca1 PCI card (instance number : 1) and disk array equipment(target = 3) being added.

```
# vi /kernel/drv/fjpfca.conf  <Return>
          :
port=
"fjpfca0:nport",
"fjpfca1:nport:sp1";               <- added

fcp-bind-target=
"fjpfca0t0:0x50000e1000002501",
"fjpfca1t3:0x50000e1000002ab0";    <- added

max-throttle=
"fjpfca0t0:128",
"fjpfca1t3:240";                   <- added
          :
```

iii. Execute the following command to make the fjpfca driver take in the added part.

```
# /usr/sbin/FJSVpfca/fc_config -v  <Return>
```

b. Add device driver configuration
Add device driver configuration with the following procedures according to the I/O device type.

[Disk/Tape devices(Solaris OE : sd/st)]
This example is a case where a disk device of target=2, lun=0 is pre-defined in procedure 1. and added in procedure 5. If the device is a tape device, re-read "sd" as "st" in this procedure and perform the same operation.

i. See /kernel/drv/sd.conf and check that target=2 and lun=0 are defined.
When this definition does not exist, add the definition. If it is commented out, remove comments.

```
Disk equipment
# vi /kernel/drv/sd.conf  <Return>
        :
name="sd" class="scsi"
     target=0 lun=0;
        :
name="sd" class="scsi"   In case the definition does not exist,
```

```
     target=2 lun=0;    add or remove comments


Tape equipment
# vi /kernel/drv/st.conf  <Return>
          :
   name="st" class="scsi"
     target=0 lun=0;
          :
   name="st" class="scsi"  In case the definition does not exist,
     target=5 lun=0;      add or remove comments
```

ii. To make the sd driver set up the disk device added, execute the following commands. When the commands are executed, the following messages are displayed but do not effect performance.
Similar messages are displayed when a tape device already exists when a tape device is added, but does not effect performance.

```
Disk device
# update_drv -f sd  <Return>
Cannot unload module: sd
Will be unloaded upon reboot.
Forcing update of sd.conf.

Tape device
# update_drv -f st  <Return>
```

iii. After saving the execution result of the prtpicl(1M) command, take the difference between the result and the information obtained in 2.a. and get the physical path name of additional disk device, then get the logical path name under /dev/dsk.
In the following example, "**c2t2d0**" is the logical path name of the disk device added.

```
# prtpicl -v > /tmp/prtpicl.post  <Return>


Disk equipment
# diff /tmp/prtpicl.pre /tmp/prtpicl.post | grep devfs-path | grep sd
<Return>
>   :devfs-path   /pci@89,4000/scsi@2/sd@2,0
# ls -l /dev/dsk | grep /pci@89,4000/scsi@2/sd@2,0  <Return>
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t2d0s0
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:a
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t2d0s1
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:b
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t2d0s2
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:c
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t2d0s3
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:d
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t2d0s4
```

```
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:e
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t2d0s5
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:f
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t2d0s6
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:g
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c2t2d0s7
-> ../../devices/pci@89,4000/scsi@2/sd@2,0:h
```

Tape equipment

```
# diff /tmp/prtpicl.pre /tmp/prtpicl.post | grep devfs-path | grep st
<Return>
>   :devfs-path   /pci@89,4000/scsi@2/st@5,0
# ls -l /dev/rmt | grep /pci@89,4000/scsi@2/st@5,0  <Return>
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1
-> ../../devices/pci@89,4000/scsi@2/st@5,0:
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1b
-> ../../devices/pci@89,4000/scsi@2/st@5,0:b
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1bn
-> ../../devices/pci@89,4000/scsi@2/st@5,0:bn
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1c
-> ../../devices/pci@89,4000/scsi@2/st@5,0:c
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1cb
-> ../../devices/pci@89,4000/scsi@2/st@5,0:cb
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1cbn
-> ../../devices/pci@89,4000/scsi@2/st@5,0:cbn
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1cn
-> ../../devices/pci@89,4000/scsi@2/st@5,0:cn
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1h
-> ../../devices/pci@89,4000/scsi@2/st@5,0:h
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1hb
-> ../../devices/pci@89,4000/scsi@2/st@5,0:hb
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1hbn
-> ../../devices/pci@89,4000/scsi@2/st@5,0:hbn
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  1hn
-> ../../devices/pci@89,4000/scsi@2/st@5,0:hn
```

iv. If the device is a disk device, start the format(1M) command and configure labeling and partition(slice) information of the logical path name of added disks.
If the device is a tape device, this procedure is not needed.

[Disk array devices(hard disk driver : hddv)]

This is an example where an disk array device at target=3, lun=0-2 is pre-defined in procedure 5. If this is the first time to pre-define a disk array device, begin with procedure 6.b.i. If this is not the first time to pre-define a disk array device, begin with procedure 6.b.v.

i. Check if /kernel/drv/hddv.conf exists

Check if /kernel/drv/hddv.conf exists. If it exists, the hard disk driver is being installed, so start from procedure 6.b.iii. If it does not exist, the system is being operated with none-active extension procedures, so start from procedure 6.b.ii.

```
 Not exist (new install)       [ to procedure 6.b.iii.]
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf: No such file or directory


Exists(operation with non-active extension procedures)  [ to procedure
6.b.ii.]
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf
```

ii. When operating with non-active extension procedures, execute the following commands and confirm that /kernel/drv/hddv.conf is deleted.

```
# hddvadm -i  <Return>
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf: No such file or directory
```

iii. Use the following command to create hddv.conf file for the active extension function. The message examples for "operation with non-active extension procedures" means that disk array device GR720 currently used with taget=1(lun=0) is registered twice on the sd/hddv driver.

```
New installation           [ to procedure 6.b.v.]
# hddvadm -M  <Return>


Operation with non-active extension procedures   [ to procedure 6.b.iv.]
# hddvadm -M  <Return>
already ID= 1 exists. /devices/pci@1d,2000/fibre-
channel@1/sd@1,0:a,raw : GR720
```

iv. When performing pre-defined extension, the same target number cannot be registered in sd.conf and hddv.conf, so delete or comment out the conflicting definition parts in sd.conf file when definition in the two files conflict with each other.
The following example shows the case that target=1 is repeated.

```
# vi /kernel/drv/sd.conf  <Return>
          :
name="sd" class="scsi"
     target=0 lun=0;
          :
#name="sd" class="scsi"
     target=1 lun=0;
#name="sd" class="scsi"    Delete or comment out
     target=1 lun=1;     the duplication definition parts in sd/hddv
#name="sd" class="scsi"
     target=1 lun=2;
          :
```

A target number that is already used by the sd driver of the system disk or CD-ROM cannot be deleted. In this case, change the target number in hddv.conf and of the disk array device

so that it does not conflict with sd.conf.

v. Edit the /kernel/drv/hddv.conf file created automatically by the hddvadm -M command, and remove comments of the LUN definition to pre-define and the LUN definition currently being used with non-active extension procedures.
If case active extension has been performed before and the LUN definition to extend the next procedures 1.f. - 1.g. are not needed. .
The following example shows target=3 , lun=0~2 for pre-defined extension and disk array device GR720 (target=1) in use with non-active extension procedures being registered.

```
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf
# vi /kernel/drv/hddv.conf  <Return>
          :
#name="hddv" class="scsi" target=0 lun=0;
          :
#name="hddv" class="scsi" target=0 lun=7;
          :
name="hddv" class="scsi" target=1 lun=0;
name="hddv" class="scsi" target=1 lun=1;  Remove comments of the LUN in use
name="hddv" class="scsi" target=1 lun=2;
          :
name="hddv" class="scsi" target=3 lun=0;
name="hddv" class="scsi" target=3 lun=1;  Remove comments of extended LUN
name="hddv" class="scsi" target=3 lun=2;
#name="hddv" class="scsi" target=3 lun=3;
          :
```

vi. Check the LUN definition overlap between sd.conf and hddv.conf by executing hddvchk_conf command.
The message example for "Check NG" means that target=3 is defined in both files, and the procedures must be performed again from procedure 6.b.iv.
The followings are examples of execution of the FJSVhddv package installation to /opt. The path of the hddvchk_conf command depends on the installed directory.

```
Check OK   [ to procedure 6.b.vii.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
/kernel/drv/sd.conf and /kernel/drv/hddv.conf seems to be correct


Check NG   [ to procedure 6.b.iv.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
NG ID = [ 3 ]
```

vii. Execute the following command in order to make the hard disk driver set up the LUN added.

```
# update_drv -f hddv  <Return>
```

In case disk array equipment currently used exists, the following messages are displayed but do not effect performance.

```
# update_drv -f hddv  <Return>
```

```
Cannot unload module: hddv
Will be unloaded upon reboot.
Forcing update of hddv.conf.
```

viii. After saving execution result of the prtpicl(1M) command, get the physical path name of added LUN by taking the difference between the result and the information obtained in 3.a, and calculate the corresponding logical path name under /dev/dsk
In the following example, "**c3t3d0**" is logical path name of added lun=0. The logical path name of lun=1,2 is set to "**c3t3d1**", "**c3t3d2**" respectively, because c**X**t**Y**d**Z**means **X**: PCI card number, **Y**: target number, **Z**: lun number .

```
# prtpicl -v > /tmp/prtpicl.post  <Return>
# diff /tmp/prtpicl.pre /tmp/prtpicl.post | grep devfs-path | grep hddv
<Return>
>    :devfs-path    /pci@89,2000/fibre-channel@1/hddv@3,0
>    :devfs-path    /pci@89,2000/fibre-channel@1/hddv@3,1
>    :devfs-path    /pci@89,2000/fibre-channel@1/hddv@3,2
# ls -l /dev/dsk | grep /pci@89,2000/fibre-channel@1/hddv@3,0  <Return>
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s0
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:a
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s1
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:b
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s2
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:c
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s3
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:d
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s4
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:e
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s5
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:f
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s6
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:g
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s7
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:h
```

ix. Start the format(1M) command and configure labeling and partition(slice) information calculated in 6.b.viii. to the logical path name of each extended LUN.

7. Setup applications
Add configuration for the application using the logical path name determined in 6.b.

[Disk device(filesystem operation)]
Create file system.
To mount at system start, add an entry in the /etc/vfstab file.

```
# newfs /dev/rdsk/c2t2d0s6  <Return>
# vi /etc/vfstab  <Return>
          :
/dev/dsk/c2t2d0s6  /dev/rdsk/c2t2d0s6   /mnt   ufs   1   yes   -
          :
```

[Disk device(raw access operation)/Tape device]
Configure applications.
For details, see the manual of each application.

[Disk device(swap device)]
Add swap device.
To make valid at system start, add an entry in the /etc/vfstab file.

```
# swap -a /dev/dsk/c2t2d0s4 <Return>
# vi /etc/vfstab <Return>
          :
/dev/dsk/c2t2d0s4   -   -   swap    -   no  -
          :
# swap -l <Return>
swapfile               dev    swaplo  blocks  free
          :
/dev/dsk/c2t2d0s4  32,164     16  788384  683680
```

8. Start applications
   Use the logical path name determined in 6.b. to start applications.

   [Disk device(filesystem operation)]
   Mount filesystem

```
# mount -F ufs /dev/dsk/c2t2d0s6 /mnt <Return>
```

   [Disk device(raw access operation)/Tape device]
   Start applications.
   For details, see the manual of each application.

   [Disk device(swap device)]
   This procedure is not required.

3.2.2.2 Extension of PCI cards with new redundant system

The procedure to add PCI cards to a new redundant system on a Solaris 9 OE system is explained with examples for the multipath software below.

- Multipath Disk Control (MPHD)
- GR Multipath Driver (GRMPD)

an example with GR multipath driver (GRMPD).

If multipath other multipath software are being used, see the manual of each product.

1. Stop the machine administration hardware monitoring daemon
   Use the following command to stop the hardware monitoring daemon of machine administration.

```
# /usr/sbin/FJSVmadm/prephp <Return>
```

2. Add PCI cards
   Add PCI cards according to the following procedure.
   To add PCI cards with the following configurations, the WWPN(World Wide Port Name: 16-digit number) of the card to add is needed.
   - If zone configuration by WWPN is done with Fibre Channel switch SN200 series.
   - If the Host Affinity function of ETERNUS3000/GR700/800 series is used.

   When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS (except for Vixel) and SP5000 SRM Facility, go to procedure 2.a, because these products get WWPNs from PCI cards automatically.
   The WWPN of the PCI card can be known from the eight characters shown on a label on the front plate of

the card. These characters represent the bottom eight digits of the WWPN in hexadecimal form. The top eight digits are fixed to 10000000 in hexadecimal form.
For example, if the following label is shown on the front plate of the card, the WWPN of the adding card is 100000000e44061.

0e24 4061

a. Before extension of PCI cards, save the current execution result of the prtpicl(1M) command (current configuration at present.)

```
# prtpicl -v > /tmp/prtpicl.pre  <Return>
```

b. Check the status of the slot.
The relationship of the position of a PCI card and format of "Ap_Id" is shown below.

Location of PCI slot	Definition of slot	Format of "Ap_Id"
Inside chassis of PRIMEPOWER 900/1500/2500	Chassis No.: **X** System board No.: **Y** Physical slot No.: **ZZ**	pcipsyN:C**X**M0**Y**-PCI#slot**ZZ**
PCI/Disk BOX	I/O Chassis No.: **X** PCI/Disk BOX No.: **Y** Physical slot No.: **ZZ**	pcipsyN:R**X**B0**Y**-PCI#slot**ZZ**
Other	Physical slot No.: **ZZ**	pcipsyN:PCI#slot**ZZ**

Below, an example of adding a PCI card to chassis number : **0**, PCI/Disk BOX number : **1**, physical slot number : **02**, slot in PCI/Disk BOX(R**0**B0**1**-PCI#slot**02**) is explained.

Check the status of the target slot is "empty unconfigured."

```
# cfgadm | grep R0B01-PCI#slot02  <Return>
pcipsy18:R0B01-PCI#slot02  unknown    empty  unconfigured  unknown
```

c. To confirm the slot position at extension operation, blink the ALARM LED of Ap_Id displayed in procedure 2.b.

```
# cfgadm -x led=fault,mode=blink pcipsy18:R0B01-PCI#slot02  <Return>
```

d. After checking that the READY LED on the PCI slot to add the PCI card is off, add PCI cards. This operation is performed by our customer support.

e. Use the cfgadm(1M) command to check that the slot status of the added PCI card changed to "disconnected unconfigured".

```
# cfgadm pcipsy18:R0B01-PCI#slot02  <Return>
Ap_Id                      Type       Receptacle    Occupant      Condition
pcipsy18:R0B01-PCI#slot02  unknown    disconnected  unconfigured  unknown
```

3. Connect the PCI card
Connect the replaced PCI card using the cfgadm(1M) command with the configure option, or by pushing the button corresponding to the extension slot position. Note that the push button is only effective in multiuser mode. After the new PCI card is connected, use the cfgadm(1M) command and confirm that the slot status has changed to "connected configured."
If a large-scale configuration of I/O devices is connected to the PCI card in the target slot, command execution for status confirmation may take time.

```
# cfgadm -c configure pcipsy18:R0B01-PCI#slot02  <Return>
# cfgadm pcipsy18:R0B01-PCI#slot02  <Return>
Ap_Id                    Type      Receptacle    Occupant    Condition
pcipsy18:R0B01-PCI#slot02   mult/hp    connected    configured    ok
```

4. Update hardware configuration information of machine administration/ Start the hardware monitoring daemon.
   Execute the following commands to update hardware configuration information of machine administration and to restart the hardware monitoring daemon.

```
# /usr/sbin/FJSVmadm/postphp  <Return>
```

5. Connect equipment to PCI cards
   Connect new disk array device in redundant configuration to the extended PCI card.
   When the Host Affinity function is being used with Fibre Channel switch SN200 series, Host Affinity must be configured for the device added.

   [When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS (except for Vixel) and SP5000 SRM Facility]
   WWPN zoning of SN200 series, Host affinity of Array disk device ETERNUS3000/GR700/800 series and binding setting for fjpfca.conf/sd.conf of host can be added from SAN management window automatically.
   See “Operation guide” or “User's Manual” of each product for details.
   Before using this function, the following procedure 6.a.i. and 6.a.ii. [except fcp-bind-target setting] are necessary.
   Moreover, begin from procedure 6.a.iii. after executing this function.

   [When not using SAN management function of the above products]
   SN200 series need to be reconfigured individually. For details, see "SN200 Series Affinity User's Guide" or "ETERNUS3000/GR700/800 series GRmgr User's Guide."

   Note:
   When changing Affinity configuration on SN200 series, I/O to other devices is effected by the change, and may result in temporal errors.
   I/O to disk array devices recovers normally because of retry processes, but on Fibre Channel tape devices, backup processes may end in errors. Stop backup before changing Affinity configuration.

6. Set up driver
   Add driver configurations with the operation below.
   With file related drivers, configuration for the PCI card driver and the device driver is needed.

   a. Add configuration for PCI card driver

      [SCSI driver(Solaris OE:glm, FUJITSU PCI SCSI Adapter Driver)]
      Configuration is not `required`.

      [Fibre Channel driver(FUJITSU PCI Fibre Channel)]

      i. After saving execution result of prtpicl (1M) command, take the difference between the result and the information obtained in procedure 2.a. to calculate the instance number of the fjpfca driver of the PCI card added.
         In the following example, instance number is 1, indicating that the interface name of the PCI card added is fjpfca1.
         Making the LED of the PCI card blink using the fc_adm command can specify the relation between the instance number and the slot position. The relation between the WWPN and instance number of the PCI card added can be known from the "fc_info -a" command.

```
# prtpicl -v > /tmp/prtpicl.pfca  <Return>
# diff /tmp/prtpicl.pre /tmp/prtpicl.pfca | more <Return>
          :
>   :status   okay
>   :fru   PCI slot(PCI#02 at Rack#0,IO_BOX#1)
>   :component-name   R0B01-PCI#02
>   :devfs-path   /pci@8d,2000/fibre-channel@1
>   :driver-name   fjpfca
>   :binding-name  pci10cf,105c
```

```
>    :bus-addr  1
>    :instance  1
>    :_class    obp-device
>    :name  fibre-channel
           :
```

ii. Edit the /kernel/drv/fjpfca.conf file, and add the following property definitions to the instance number determined in 6.a.i.

port(type of topology, link speed)
Defined when adding PCI cards with Fabric connection.
This definition is not needed with FC-AL connection.

fcp-bind-target(connection between target ID and WWN)
Defined when adding disk array with Fabric connection.
This definition is not needed with FC-AL connection.

max-throttle(maximum issued SCSI commands)
Defined to limit the number of SCSI commands issued at the same time.
The need and value of this definition depends on the disk array device, so refer to the manuals of each disk array device. Below is an example where definitions for a PCI card at fjpfca1(instance number 1) and a disk array device at target=3 is added.

```
# vi /kernel/drv/fjpfca.conf  <Return>
           :
port=
"fjpfca0:nport",
"fjpfca1:nport:sp1";                <- add

fcp-bind-target=
"fjpfca0t0:0x50000e1000002501",
"fjpfca1t3:0x50000e1000002ab0";    <- add

max-throttle=
"fjpfca0t0:128",
"fjpfca1t3:240";                   <- add
           :
```

iii. Execute the following command to make the fjpfca driver set up the additional part.

```
# /usr/sbin/FJSVpfca/fc_config -v  <Return>
```

b. Add configuration of device driver
This is an example where an disk array device in redundant configuration at target=3, lun=0-2 is added in procedure 5.
If this is the first time to pre-define a disk array device, begin with procedure 6.b.i. If this is not the first time to pre-define a disk array device, begin with procedure 6.b.v.

i. Check if /kernel/drv/hddv.conf exists
Check if /kernel/drv/hddv.conf exists. If it exists, the hard disk driver is being installed, so start from procedure 6.b.iii. If it does not exist, the system is being operated with none-active extension procedures, so start from procedure 6.b.ii.

```
Not exist(new install)      [ to Procedure 6.b.iii.]
# ls /kernel/drv/hddv.conf  <Return>
```

```
/kernel/drv/hddv.conf: No such file or directory


Exist(operation with non-active extension procedures) [ to Procedure
6.b.ii. ]
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf
```

ii. When operating with non-active extension procedures, execute the following commands and confirm that /kernel/drv/hddv.conf is deleted.

```
# hddvadm -i  <Return>
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf: No such file or directory
```

iii. Use the following command to create hddv.conf file for the pre-defined extension function. The message examples for "operation with non-active extension procedures" means that disk array device GR720 currently used with taget=1(lun=0) is registered twice on the sd/hddv driver.

```
New installation            [ to procedure 6.b.v.]
# hddvadm -M  <Return>


Operation with non-active extension procedures   [ to procedure 6.b.iv.]
# hddvadm -M  <Return>
already ID= 1 exists. /devices/pci@1d,2000/fibre-
channel@1/sd@1,0:a,raw : GR720
```

iv. .When performing active extension, the same target number cannot be registered in sd.conf and hddv.conf, so delete or comment out the conflicting definition parts in sd.conf file when definition in the two files conflict with each other.
The following example shows the case that target=1 is repeated

```
# vi /kernel/drv/sd.conf  <Return>
          :
name="sd" class="scsi"
     target=0 lun=0;
          :
#name="sd" class="scsi"
     target=1 lun=0;
#name="sd" class="scsi"    delete or comment out
     target=1 lun=1;     the duplication definition parts in sd/hddv
#name="sd" class="scsi"
     target=1 lun=2;
          :
```

A target number that is already used by the sd driver of the system disk or CD-ROM cannot be deleted. In this case, change the target number in hddv.conf and of the disk array device so that it does not conflict with sd.conf.

v. Edit the /kernel/drv/hddv.conf file created automatically by the hddvadm -M command, and remove comments of the LUN definition to pre-define and the LUN definition currently being used with non-active extension procedures.
If case active extension has been performed before and the LUN definition to extend the

next procedures 1.f. - 1.g. are not needed.
The following example shows target=3, lun=0~2 for pre-defined extension and disk array device GR720 (target=1) in use with non-active extension procedures being registered.

```
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf
# vi /kernel/drv/hddv.conf  <Return>
          :
#name="hddv" class="scsi" target=0 lun=0;
          :
#name="hddv" class="scsi" target=0 lun=7;
          :
name="hddv" class="scsi" target=1 lun=0;
name="hddv" class="scsi" target=1 lun=1;  remove comments of the LUN in use
name="hddv" class="scsi" target=1 lun=2;
          :
name="hddv" class="scsi" target=3 lun=0;
name="hddv" class="scsi" target=3 lun=1;  remove comments of extended LUN
name="hddv" class="scsi" target=3 lun=2;
#name="hddv" class="scsi" target=3 lun=3;
          :
```

vi. Check the LUN definition overlap between sd.conf and hddv.conf by executing hddvchk_conf command.
The message example for "Check NG" means that target=3 is defined in both files, and the procedures must be performed again from procedure 6.b.iv.
The followings are examples of execution of the FJSVhddv package installation to /opt. The path of the hddvchk_conf command depends on the installed directory.

```
Check OK   [ to procedure 6.b.vii.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
/kernel/drv/sd.conf and /kernel/drv/hddv.conf seems to be correct


Check NG   [ to procedure 6.b.iv]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
NG ID = [ 3 ]
```

vii. Execute the following command in order to make the hard disk driver set up the LUN added.

```
# update_drv -f hddv  <Return>
```

In case disk array equipment currently used exists, the following messages are displayed but do not effect performance.

```
# update_drv -f hddv  <Return>
Cannot unload module: hddv
Will be unloaded upon reboot.
Forcing update of hddv.conf.
```

viii. After saving execution result of the prtpicl(1M) command, get the physical path name of

added LUN by taking the difference between the result and the information obtained in 2.a, and calculate the corresponding logical path name under /dev/dsk
In the following example, "**c3t3d0**" is logical path name of added lun=0. The logical path name of lun=1,2 is set to "**c3t3d1**", "**c3t3d2**" respectively, because c**X**t**Y**d**Z**means **X**: PCI card number, **Y**: target number, **Z**: lun number

```
# prtpicl -v > /tmp/prtpicl.post  <Return>
# diff /tmp/prtpicl.pre /tmp/prtpicl.post | grep devfs-path | grep hddv
<Return>
>    :devfs-path    /pci@89,2000/fibre-channel@1/hddv@3,0
>    :devfs-path    /pci@89,2000/fibre-channel@1/hddv@3,1
>    :devfs-path    /pci@89,2000/fibre-channel@1/hddv@3,2
# ls -l /dev/dsk | grep /pci@89,2000/fibre-channel@1/hddv@3,0  <Return>
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s0
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:a
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s1
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:b
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s2
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:c
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s3
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:d
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s4
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:e
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s5
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:f
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s6
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:g
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s7
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:h
```

ix. Start the format(1M) command and configure labeling and partition(slice) information calculated in 6.b.viii. to the logical path name of each extended LUN.

7. Setup redundant system
Execute the following commands to create a new redundant system.
In case of MPHD, the commands need to be executed for each LUN, specifying the path name determined in 6.b.viii as a parameter.
The italic letters below are the new multipath names, and the underlined letters are the path names that compose the multipath.

```
MPHD
# /usr/opt/FJSViomp/bin/iompadm -c mphd new /dev/rdsk/c3t3d0s2 /dev/rdsk/c4t3d0s2
<Return>
IOMP: /dev/FJSVmphd/fiomp/adm2
Node:
       /dev/FJSVmphd/rdsk/mphd2s0
       /dev/FJSVmphd/rdsk/mphd2s1
       /dev/FJSVmphd/rdsk/mphd2s2
       /dev/FJSVmphd/rdsk/mphd2s3
       /dev/FJSVmphd/rdsk/mphd2s4
```

```
       /dev/FJSVmphd/rdsk/mphd2s5
       /dev/FJSVmphd/rdsk/mphd2s6
       /dev/FJSVmphd/rdsk/mphd2s7

GRMPD
# /usr/sbin/mplbconfig -nv  <Return>
*** Phase 1: checking /dev/rdsk ***
*** Phase 2: checking multi path ***
*** Phase 3: analyzing path ***
     Path    : Action : Element path          : LUN : Storage        : Zone
     mplb0   : exist  : c1t4d0s2 c2t3d0s2      : 0 : GR720464f-12340001 :
       :
     mplb8   : new    : c3t3d0s2 c4t3d0s2      : 0 : GR8404550-  000213 :
       :
# /usr/sbin/mplbconfig -av  <Return>
             :
IOMP: /dev/FJSVmplb/fiomp/adm8
Node:
       /dev/FJSVmplb/rdsk/mplb8s0
       /dev/FJSVmplb/rdsk/mplb8s1
       /dev/FJSVmplb/rdsk/mplb8s2
       /dev/FJSVmplb/rdsk/mplb8s3
       /dev/FJSVmplb/rdsk/mplb8s4
       /dev/FJSVmplb/rdsk/mplb8s5
       /dev/FJSVmplb/rdsk/mplb8s6
       /dev/FJSVmplb/rdsk/mplb8s7
             :
```

8. Setup redundant system
   Add configuration for the application using the logical path name determined in 7.

[filesystem operation]
Create file system.
To mount at system start, add an entry in the /etc/vfstab file.

```
# newfs /dev/FJSVmphd/rdsk/mphd2s6  <Return>
# vi /etc/vfstab  <Return>
          :
/dev/FJSVmphd/dsk/mphd2s6  /dev/FJSVmphd/rdsk/mphd2s6   /mnt   ufs   1   yes   -
          :
```

[raw access operation]
Configure applications.
For details, see the manual of each application.

[swap device]
Add swap device.
To make valid at system start, add an entry in the /etc/vfstab file.

```
# swap -a /dev/FJSVmphd/dsk/mphd2s4  <Return>
# vi /etc/vfstab  <Return>
          :
/dev/FJSVmphd/dsk/mphd2s4-   -   swap    -   no  -
          :
# swap -l  <Return>
swapfile               dev    swaplo  blocks  free
          :
/dev/FJSVmphd/dsk/mphd2s4  32,164     16  788384  683680
```

9. Start applications
   Use the logical path name determined in 7. to start applications.

   [filesystem operation]
   Mount filesystem

```
# mount -F ufs /dev/FJSVmphd/dsk/mphd2s6 /mnt  <Return>
```

   [raw access operation]
   Start applications.
   For details, see the manual of each application.

   [swap device]
   This procedure is not required.

3.2.2.3 Extension of PCI cards to redundant system

The procedure to add PCI cards to a redundant system in order to extend redundancy on a Solaris 9 OE system is explained with an example with GR multipath driver (GRMPD).

There is no additional pattern for Multipath Disk Control (MPHD).
If multipath software products other than GRMPD are being used, see the manual of each product.

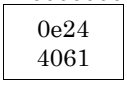
No configuration is needed for applications when adding PCI cards to an existing redundant system.

1. Stop the machine administration hardware monitoring daemon
   Use the following command to stop the hardware monitoring daemon of machine administration.

```
# /usr/sbin/FJSVmadm/prephp  <Return>
```

2. Add PCI cards
   To add PCI cards with the following configurations, the WWPN(World Wide Port Name: 16-digit number) of the card to add is needed.
   - If zone configuration by WWPN is done with Fibre Channel switch SN200 series.
   - If the Host Affinity function of ETERNUS3000/GR700/800 series is used.

   When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS (except for Vixel) and SP5000 SRM Facility, go to procedure 2.a, because these products get WWPNs from PCI cards automatically.
   The WWPN of the PCI card can be known from the eight characters shown on a label on the front plate of the card. These characters represent the bottom eight digits of the WWPN in hexadecimal form. The top eight digits are fixed to 10000000 in hexadecimal form.
   For example, if the following label is shown on the front plate of the card, the WWPN of the adding card is 100000000e44061.



   a. Before extension of PCI cards, save the current execution result of the prtpicl(1M) command (current configuration at present.)

```
# prtpicl -v > /tmp/prtpicl.pre  <Return>
```

b. Check the status of the slot.
When adding PCI cards to a redundant configuration, several PCI cards may be added to the system. In this case, perform procedures to 3. on each card.
The relationship of the position of a PCI card and format of "Ap_Id" is shown below.

Location of PCI slot	Definition of slot	Format of "Ap_Id"
Inside chassis of PRIMEPOWER 900/1500/2500	Chassis No.: **X** System board No.: **Y** Physical slot No.: **ZZ**	pcipsyN:C**X**M0**Y**-PCI#slot**ZZ**
PCI/Disk BOX	I/O Chassis No.: **X** PCI/Disk BOX No.: **Y** Physical slot No.: **ZZ**	pcipsyN:R**X**B0**Y**-PCI#slot**ZZ**
Other	Physical slot No.: **ZZ**	pcipsyN:PCI#slot**ZZ**

Below, an example of adding a PCI card to chassis number : **0**, PCI/Disk BOX number : **1**, physical slot number : **02**, slot in PCI/Disk BOX(R**0**B0**1**-PCI#slot**02**) is explained.

Check the status of the target slot is "empty unconfigured."

```
# cfgadm | grep R0B01-PCI#slot02  <Return>
pcipsy18:R0B01-PCI#slot02  unknown    empty  unconfigured  unknown
```

c. To confirm the slot position at extension operation, blink the ALARM LED of Ap_Id displayed in procedure 2.b.

```
# cfgadm -x led=fault,mode=blink pcipsy18:R0B01-PCI#slot02  <Return>
```

d. After checking that the READY LED on the PCI slot to add the PCI card is off, add PCI cards. This operation is performed by our customer support.

e. Use the cfgadm(1M) command to check that the slot status of the added PCI card changed to "disconnected unconfigured".

```
# cfgadm pcipsy18:R0B01-PCI#slot02  <Return>
Ap_Id                       Type       Receptacle    Occupant      Condition
pcipsy18:R0B01-PCI#slot02  unknown    disconnected  unconfigured  unknown
```

3. Connect the PCI card
Connect the replaced PCI card using the cfgadm(1M) command with the configure option, or by pushing the button corresponding to the extension slot position. Note that the push button is only effective in multiuser mode. After the new PCI card is connected, use the cfgadm(1M) command and confirm that the slot status has changed to "connected configured."
If a large-scale configuration of I/O devices is connected to the PCI card in the target slot, command execution for status confirmation may take time.

```
# cfgadm -c configure pcipsy18:R0B01-PCI#slot02  <Return>
# cfgadm pcipsy18:R0B01-PCI#slot02  <Return>
Ap_Id                       Type      Receptacle    Occupant    Condition
pcipsy18:R0B01-PCI#slot02   mult/hp   connected     configured   ok
```

4. Update hardware configuration information of machine administration/ Start the hardware monitoring daemon.
Execute the following commands to update hardware configuration information of machine administration and to restart the hardware monitoring daemon.

```
# /usr/sbin/FJSVmadm/postphp  <Return>
```

5. Connect equipment to PCI cards

Connect new disk array device in redundant configuration to the extended PCI card.
When the Host Affinity function is being used with Fibre Channel switch SN200 series, Host Affinity must be configured for the device added.

[When using any of SAN management function of Systemwalker/StorageMGR, SANView for ETERNUS (except for Vixel) and SP5000 SRM Facility]
WWPN zoning of SN200 series, Host affinity of Array disk device ETERNUS3000/GR700/800 series and binding setting for fjpfca.conf/sd.conf of host can be added from SAN management window automatically.
See “Operation guide” or “User's Manual” of each product for details.
Before using this function, the following procedure 6.a.i. and 6.a.ii. [except fcp-bind-target setting] are necessary.
Moreover, begin from procedure 6.a.iii. after executing this function.

[When not using SAN management function of the above products]
SN200 series need to be reconfigured individually. For details, see "SN200 Series Affinity User's Guide" or "ETERNUS3000/GR700/800 series GRmgr User's Guide."

Note:
When changing Affinity configuration on SN200 series, I/O to other devices is effected by the change, and may result in temporal errors.
I/O to disk array devices recovers normally because of retry processes, but on Fibre Channel tape devices, backup processes may end in errors. Stop backup before changing Affinity configuration.

6. Set up driver
   Add driver configurations with the operation below.
   With file related drivers, configuration for the PCI card driver and the device driver is needed.

   a. Add configuration for PCI card driver

      [SCSI driver(Solaris OE:glm, FUJITSU PCI SCSI Adapter Driver)]
      Configuration is not `required`.

      [Fibre Channel driver(FUJITSU PCI Fibre Channel)]

      i. After saving execution result of prtpicl (1M) command, take the difference between the result and the information obtained in procedure 2.a. to calculate the instance number of the fjpfca driver of the PCI card added.
      In the following example, instance number is 1, indicating that the interface name of the PCI card added is fjpfca1.
      Making the LED of the PCI card blink using the fc_adm command can specify the relation between the instance number and the slot position. The relation between the WWPN and instance number of the PCI card added can be known from the "fc_info -a" command.

```
# prtpicl -v > /tmp/prtpicl.pfca  <Return>
# diff /tmp/prtpicl.pre /tmp/prtpicl.pfca | more <Return>
          :
>   :status   okay
>   :fru   PCI slot(PCI#02 at Rack#0,IO_BOX#1)
>   :component-name   R0B01-PCI#02
>   :devfs-path   /pci@8d,2000/fibre-channel@1
>   :driver-name   fjpfca
>   :binding-name  pci10cf,105c
>   :bus-addr  1
>   :instance  1
>   :_class   obp-device
>   :name  fibre-channel
          :
```

      ii. Edit the /kernel/drv/fjpfca.conf file, and add the following property definitions to the

instance number determined in 6.a.i.

port(type of topology, link speed)
Defined when adding PCI cards with Fabric connection.
This definition is not needed with FC-AL connection.

fcp-bind-target(connection between target ID and WWN)
Defined when adding disk array with Fabric connection.
This definition is not needed with FC-AL connection.

max-throttle(maximum issued SCSI commands)
Defined to limit the number of SCSI commands issued at the same time.
The need and value of this definition depends on the disk array device, so refer to the manuals of each disk array device.
Below is an example where definitions for a PCI card at fjpfca1(instance number 1) and a disk array device at target=3 is added.

```
# vi /kernel/drv/fjpfca.conf  <Return>
          :
port=
"fjpfca0:nport",
"fjpfca1:nport:sp1";               <- add


fcp-bind-target=
"fjpfca0t0:0x50000e1000002501",
"fjpfca1t3:0x50000e1000002ab0";    <- add


max-throttle=
"fjpfca0t0:128",
"fjpfca1t3:240";                   <- add
          :
```

iii. Execute the following command to make the fjpfca driver set up the additional part.

```
# /usr/sbin/FJSVpfca/fc_config -v  <Return>
```

b. Add configuration of device driver
This is an example where a disk array device in redundant configuration at target=3, lun=0-2 is added in procedure 5.
If this is the first time to pre-define a disk array device, begin with procedure 6.b.i. If this is not the first time to pre-define a disk array device, begin with procedure 6.b.v.

i. Check if /kernel/drv/hddv.conf exists.
Check if /kernel/drv/hddv.conf exists. If it exists, the hard disk driver is being installed, so start from procedure 6.b.iii. If it does not exist, the system is being operated with none-active extension procedures, so start from procedure 6.b.ii.

```
Not exist(new install)      [ to procedure 6.b.iii.]
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf: No such file or directory


Exist(operation with non-active extension procedures) [ to procedure
6.b.ii.]
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf
```

ii. When operating with non-active extension procedures, execute the following commands and confirm that /kernel/drv/hddv.conf is deleted.

```
# hddvadm -i  <Return>
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf: No such file or directory
```

iii. Use the following command to create hddv.conf file for the pre-defined extension function. The message examples for "operation with non-active extension procedures" means that disk array device GR720 currently used with taget=1(lun=0) is registered twice on the sd/hddv driver.

```
New installation            [ to procedure 6.b.v.]
# hddvadm -M  <Return>


Set up with a non-active extension procedure   [ to procedure 6.b.iv.]
# hddvadm -M  <Return>
already ID= 1 exists. /devices/pci@1d,2000/fibre-
channel@1/sd@1,0:a,raw : GR720
```

iv. In activity extension, because same target number can't be registered in sd.conf and hddv.conf, delete or comment out the duplication definition parts in sd.conf file when both definition are overlapped.The following example shows the case that target=1 is repeated.

```
# vi /kernel/drv/sd.conf  <Return>
          :
name="sd" class="scsi"
     target=0 lun=0;
          :
#name="sd" class="scsi"
     target=1 lun=0;
#name="sd" class="scsi"    delete or comment out
     target=1 lun=1;     the duplication definition parts in sd/hddv
#name="sd" class="scsi"
     target=1 lun=2;
          :
```

A target number that is already used by the sd driver of the system disk or CD-ROM cannot be deleted. In this case, change the target number in hddv.conf and of the disk array device so that it does not conflict with sd.conf.

v. Edit the /kernel/drv/hddv.conf file created automatically by the hddvadm -M command, and remove comments of the LUN definition to pre-define and the LUN definition currently being used with non-active extension procedures.
If case active extension has been performed before and the LUN definition to extend the next procedures 1.f. - 1.g. are not needed.
The following example shows target=3 , lun=0~2 for pre-defined extension and disk array device GR720 (target=1) in use with non-active extension procedures being registered.

```
# ls /kernel/drv/hddv.conf  <Return>
/kernel/drv/hddv.conf
# vi /kernel/drv/hddv.conf  <Return>
          :
```

```
#name="hddv" class="scsi" target=0 lun=0;
          :
#name="hddv" class="scsi" target=0 lun=7;
          :
name="hddv" class="scsi" target=1 lun=0;
name="hddv" class="scsi" target=1 lun=1;  remove comments of the LUN in use
name="hddv" class="scsi" target=1 lun=2;
          :
name="hddv" class="scsi" target=3 lun=0;
name="hddv" class="scsi" target=3 lun=1;  remove comments of extended LUN
name="hddv" class="scsi" target=3 lun=2;
#name="hddv" class="scsi" target=3 lun=3;
          :
```

vi. Check the LUN definition overlap between sd.conf and hddv.conf by executing hddvchk_conf command.
The message example for "Check NG" means that target=3 is defined in both files, and the procedures must be performed again from procedure 6.b.iv.
The followings are examples of execution of the FJSVhddv package installation to /opt. The path of the hddvchk_conf command depends on the installed directory.

```
Check OK   [ to procedure 6.b.vii.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
/kernel/drv/sd.conf and /kernel/drv/hddv.conf seems to be correct


Check NG   [ to procedure 6.b.iv.]
# /opt/FJSVhddv/bin/hddvchk_conf  <Return>
NG ID = [ 3 ]
```

vii. Execute the following command in order to make the hard disk driver set up the LUN added.

```
# update_drv -f hddv  <Return>
```

In case disk array equipment currently used exists, the following messages are displayed but do not effect performance.

```
# update_drv -f hddv  <Return>
Cannot unload module: hddv
Will be unloaded upon reboot.
Forcing update of hddv.conf.
```

viii. After saving execution result of the prtpicl(1M) command, get the physical path name of added LUN by taking the difference between the result and the information obtained in 2.a, and calculate the corresponding logical path name under /dev/dsk
In the following example, "**c3t3d0**" is logical path name of added lun=0. The logical path name of lun=1,2 is set to "**c3t3d1**", "**c3t3d2**" respectively, because c**X**t**Y**d**Z**means **X**: PCI card number, **Y**: target number, **Z**: lun number

```
# prtpicl -v > /tmp/prtpicl.post  <Return>
# diff /tmp/prtpicl.pre /tmp/prtpicl.post | grep devfs-path | grep hddv
<Return>
```

```
>   :devfs-path   /pci@89,2000/fibre-channel@1/hddv@3,0
>   :devfs-path   /pci@89,2000/fibre-channel@1/hddv@3,1
>   :devfs-path   /pci@89,2000/fibre-channel@1/hddv@3,2
# ls -l /dev/dsk | grep /pci@89,2000/fibre-channel@1/hddv@3,0  <Return>
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:a
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s1
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:b
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s2
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:c
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s3
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:d
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s4
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:e
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s5
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:f
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s6
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:g
lrwxrwxrwx   1  root   root       41  Sep 20  22:53  c3t3d0s7
-> ../../devices/pci@89,2000/fibre-channel@1/hddv@3,0:h
```

ix. Start format command and configure labeling and partition (slice) information calculated at 6.b.viii to the logical path name of each extended LUN.

7. Connect to redundant system
   To connect the added PCI card to the redundant system, execute the following command and confirm that the path name determined in procedure 6.b.viii (in underlined letters) is added.

```
# /usr/sbin/mplbconfig -nv  <Return>
*** Phase 1: checking /dev/rdsk ***
*** Phase 2: checking multi path ***
*** Phase 3: analyzing path ***
     Path    : Action : Element path          : LUN : Storage        : Zone
     mplb0   : exist  : c1t4d0s2 c2t3d0s2       : 0 : GR720464f-12340001 :
       :
     mplb8   : add    : c3t3d0s2              : 0 : GR8404550-  000213 :
       :
# /usr/sbin/mplbconfig -av  <Return>
             :
IOMP: /dev/FJSVmplb/fiomp/adm8
Node:
       /dev/FJSVmplb/rdsk/mplb8s0
       /dev/FJSVmplb/rdsk/mplb8s1
       /dev/FJSVmplb/rdsk/mplb8s2
       /dev/FJSVmplb/rdsk/mplb8s3
       /dev/FJSVmplb/rdsk/mplb8s4
```

```
        /dev/FJSVmplb/rdsk/mplb8s5
        /dev/FJSVmplb/rdsk/mplb8s6
        /dev/FJSVmplb/rdsk/mplb8s7
               :
```

Chapter 4 Network devices

This chapter describes the detailed procedure about replacement or extension of network-related PCI cards by PCI Hot Plug.

Please refer to manuals of each software about the details of the redundant software.

4.1 Replacement of PCI card

This section describes replacement procedure of the PCI card by using PCI Hot Plug on non-redundant system and redundant system.

4.1.1 Replacement of PCI cards on non-redundant system

When exchanging PCI cards without redundant configuration software, you must stop applications which using PCI cards. The replacement procedure of PCI cards is as follows.

1. Stop the hardware monitoring of the Machine Administration
   The hardware monitoring of the Machine Administration is stopped by using the following command.
   Please refer to "PCI Hot Plug User's Guide" about the details of prephp (1M) command.

```
# /usr/sbin/FJSVmadm/prephp <Return>
```

2. Specify the replacing PCI card
   The interfaces (e.g. hme1) which stop applications for replacement of PCI cards are specified from warning messages outputted the console.

3. Stop applications
   By following operations, the high layer product (e.g. service, application) which uses PCI cards for exchange is stopped.
   a. When using TCP/IP interface
      The example in the case of using TCP/IP interface is shown below. Please refer to the online manual etc. about the details of the ifconfig (1M) command.
      Moreover, when high layer programs which using the TCP/IP interface needs to be stopped, please stop each high layer program. Please refer to each manual about the stop procedure of high layer programs.
      i. When using Ethernet devices

```
# /usr/sbin/ifconfig hme1 down <Return>
# /usr/sbin/ifconfig hme1 unplumb <Return>
```

      ii. When using ATM devices

```
# /opt/FJSVatm/bin/atmunlink adapter_number <Return>
```

   b. When using FNA/SNA communication protocol over Ethernet
      When not using the redundant configuration software, it is necessary to stop all high layer applications that are using Netcompo FNA-LAN, and all interfaces of Netcompo FNA-LAN.
      For details, please refer to the manual of Netcompo FNA-LAN and high layer applications.

4. Disconnect the PCI card
   a. The "Ap_Id" is displayed using the following command, based on the interface (e.g.: hme1) specified in Procedure "2. Specify the replacing PCI card." The attachment pint identifier "Ap_Id" is a value for the cfgadm (1M) command specifying the slot position of the PCI card. Please refer to "PCI Hot Plug user's guide" about the details of cfgadm (1M) and "Ap_Id."

```
# /usr/sbin/FJSVmadm/inst2comp hme1 <Return>
pcipsy3:COM00-PCI#slot02
```

   b. Obtained "Ap_Id" is specified as an argument of the cfgadm (1M) command, and it confirms that the slot status of the PCI card that disconnects is "connected configured."

```
# cfgadm pcipsy3:COM00-PCI#slot02 <Return>
Ap_Id                    Type     Receptacle  Occupant  Condition
```

```
pcipsy3:COM00-PCI#slot02   pci-pci/hp  connected  configured  ok
```

c. Please disconnect the PCI card by executing the cfgadn (1M) command with "Ap_Id," and confirm that the slot status is "disconnected unconfigured."

```
# cfgadm -c disconnect pcipsy3:COM00-PCI#slot02 <Return>
# cfgadm pcipsy3:COM00-PCI#slot02 <Return>
Ap_Id                  Type     Receptacle    Occupant    Condition
pcipsy3:COM00-PCI#slot02  pci-pci/hp  disconnected  unconfigured  unknown
```

d. To confirm the slot position at replacement operation, blink the ALARM LED of Ap_Id displayed in procedure 4.a.

```
# cfgadm –x led=fault,mode=blink pcipsy3:COM00-PCI#slot02 <Return>
```

5. Replace the PCI card
   The PCI card disconnected in procedure "4. Disconnect the PCI card" is exchanged for a new PCI card. Our customer service works about this procedure.

6. Connect the PCI card
   The cfgadm command is executed with the "configure" option to "Ap_Id," or the push button of the PCI bus slot is pushed. Then a new PCI card is connected.
   In addition, a push button is valid only in multiple user mode.
   Please confirm that the slot status is "connected configured" by using the cfgadm (1M) command after the above-mentioned procedure.

   Notes:
   If the setup of driver.conf(4) file exist, please refer to "5.1.1 Network driver configuration changing."

```
# cfgadm -c configure pcipsy3:COM00-PCI#slot02 <Return>
# cfgadm pcipsy3:COM00-PCI#slot02 <Return>
Ap_Id                  Type      Receptacle  Occupant   Condition
pcipsy3:COM00-PCI#slot02  pci-pci/hp   connected  configured  ok
```

7. Start applications
   Please restart the stopped high layer product (service, application) by the following operations.
   a. When using TCP/IP interface
      The example in the case of using TCP/IP interface is shown below.
      Furthermore, when the high layer program which is using the PCI card needs to restart, please refer to the manual of each high layer program.
      i. When using Ethernet devices

```
# /usr/sbin/ifconfig hme1 plumb <Return>
# /usr/sbin/ifconfig hme1 inet 192.168.10.10 netmask + broadcast + -
trailers up <Return>
```

      ii. When using ATM devices

```
# /opt/FJSVatm/bin/atmlink adapter_number <Return>
```

   b. When using FNA/SNA communication protocol over Ethernet
      When not using the redundant configuration software, it is necessary to start all interfaces of Netcompo FNA-LAN, and all high layer applications that are using Netcompo FNA-LAN.
      For details, please refer to the manual of Netcompo FNA-LAN and high layer applications.

8. Start the hardware monitoring of the Machine Administration
   Please execute the following commands in order to update configuration information, and restart the hardware monitoring of the Machine Administration.
   Please refer to "PCI Hot Plug User's Guide" about details of postphp (1M) command.

```
# /usr/sbin/FJSVmadm/postphp <Return>
```

4.1.2 Replacement of PCI cards on redundant system

When exchanging PCI cards with redundant configuration software, it can exchange PCI cards without stopping applications.

This subsection describes replacement procedure of PCI cards with the following redundant configuration software.

- SynfinityLink 2.2
- PRIMECLUSTER GLS 4.0 or later

The replacement and extension of PCI cards are supported only on Redundant Line Control Function of SynfinityLink.

When using other redundant software, please refer to manuals of each product.

1. Stop the hardware monitoring of the Machine Administration
   The hardware monitoring of the Machine Administration is stopped by using the following command.

```
# /usr/sbin/FJSVmadm/prephp <Return>
```

2. Specify the replacing PCI card
   The interfaces (e.g. hme1) which stop applications for replacement of PCI cards are specified from warning messages outputted the console.

3. Disconnect from redundant system
   In order to disconnect the PCI card for exchange from a redundant system, please execute the following command with the interface name specified by procedure "2. Specify the replacing PCI card."
   On the redundant system of standby mode, when an online communication path is disconnected, a standby communication path will be online communication path automatically.
   When using the GS/SURE linkage mode in Redundant Line Control Function of SynfinityLink or PRIMECLUSTER GLS, it is necessary to deactivate virtual interfaces (e.g. sha0) used by GS/SURE linkage mode.

```
Fast switching mode in Redundant Line Control Function
# /opt/FJSVhanet/usr/sbin/hanetnic delete -n sha0 -i hme1 <Return>
# /usr/sbin/ifconfig hme1 unplumb <Return>

NIC switching mode in Redundant Line Control Function
# /opt/FJSVhanet/usr/sbin/stpptl -n sha1 <Return>
# /opt/FJSVhanet/usr/sbin/hanetnic change -n sha0 <Return>

GS/SURE linkage mode in Redundant Line Control Function
# /opt/FJSVhanet/usr/sbin/stphanet -n sha0 <Return>

Multipath Function
# /usr/opt/FJSViomp/bin/iompadm -c mpnet stop mpnet0 hme1 <Return>
# /usr/opt/FJSViomp/bin/iompadm -c mpnet disconnect mpnet0 hme1 <Return>
```

4. Disconnect the PCI card
   a. Please specify the PCI bus slot position by the interface (e.g. hme1) in procedure "2. Specify the replacing PCI card."

```
# /usr/sbin/FJSVmadm/inst2comp hme1 <Return>
pcipsy21:R0B01-PCI#slot03
```

b. Obtained "Ap_Id" is specified as an argument of the cfgadm (1M) command, and it confirms that the slot status of the PCI card that disconnects is "connected configured."

```
# cfgadm pcipsy21:R0B01-PCI#slot03 <Return>
Ap_Id                        Type         Receptacle   Occupant     Condition
pcipsy21:R0B01-PCI#slot03    pci-pci/hp    connected    configured    ok
```

c. Please disconnect the PCI card by executing the cfgadn (1M) command with "Ap_Id," and confirm that the slot status is "disconnected unconfigured."

```
# cfgadm -c disconnect pcipsy21:R0B01-PCI#slot03 <Return>
# cfgadm pcipsy21:R0B01-PCI#slot03 <Return>
Ap_Id                       Type        Receptacle    Occupant     Condition
pcipsy21:R0B01-PCI#slot03  pci-pci/hp  disconnected  unconfigured unknown
```

d. To confirm the slot position at replacement operation, blink the ALARM LED of Ap_Id displayed in procedure 4.a.

```
# cfgadm -x led=fault,mode=blink pcipsy21:R0B01-PCI#slot03 <Return>
```

5. Replace the PCI card
The PCI card disconnected in procedure "4. Disconnect the PCI card" is exchanged for a new PCI card. Out customer support works about this procedure.
Notes:
When using the PRIMECLUSTER GLS Multipath Function, please exchange the PCI card after waiting 10 seconds or more, after Procedure "4. Disconnect the PCI card."

6. Connect the PCI card
The cfgadm command is executed with the "configure" option to "Ap_Id," or the push button of the PCI bus slot is pushed. Then a new PCI card is connected.
In addition, a push button is valid only in multiple user mode.
Please confirm that the slot status is "connected configured" by using the cfgadm (1M) command after the above-mentioned procedure.

Notes:
If the setup of driver.conf(4) file exist, please refer to "5.1.1 Network driver configuration changing."

```
# cfgadm -c configure pcipsy21:R0B01-PCI#slot03 <Return>
# cfgadm pcipsy21:R0B01-PCI#slot03 <Return>
Ap_Id                   Type       Receptacle    Occupant     Condition
pcipsy21:R0B01-PCI#slot03    pci-pci/hp    connected     configured    ok
```

7. Connect to redundant system
In order to connect the exchanged PCI card into a redundant system, please execute the following command with interface name specified at procedure "2. Specify the replacing PCI card."

```
Fast switching mode in Redundant Line Control Function
# /usr/sbin/ifconfig hme1 plumb <Return>
# /usr/sbin/ifconfig hme1 192.168.10.10 netmask + broadcast + -trailers up <Return>
# /opt/FJSVhanet/usr/sbin/hanetnic add -n sha0 -i hme1 <Return>

NIC switching mode in Redundant Line Control Function
# /opt/FJSVhanet/usr/sbin/hanetnic change -n sha0 <Return>
# /opt/FJSVhanet/usr/sbin/strptl -n sha1 <Return>
```

```
GS/SURE linkage mode in Redundant Line Control Function
# /opt/FJSVhanet/usr/sbin/strhanet -n sha0 <Return>


Multipath Function
# /usr/opt/FJSViomp/bin/iompadm -c mpnet connect mpnet0 hme1 <Return>
# /usr/opt/FJSViomp/bin/iompadm -c mpnet start mpnet0 hme1 <Return>
```

8. Switch back the redundant path
   Please switch back an online communication path if needed.

9. Start the hardware monitoring of the Machine Administration
   Please execute the following commands in order to update configuration information, and restart the hardware monitoring of the Machine Administration.

```
# /usr/sbin/FJSVmadm/postphp <Return>
```

4.2 Extension of PCI cards

This section describes extension procedure of the PCI card by using PCI Hot Plug on non-redundant system and redundant system.

4.2.1 Extension of PCI cards on non-redundant system

This subsection describes extension procedure of PCI cards without the redundant configuration software.

1. Stop the hardware monitoring of the Machine Administration
   The hardware monitoring of the Machine Administration is stopped by using the following command.

```
# /usr/sbin/FJSVmadm/prephp <Return>
```

2. Add PCI cards
   a. Before extending the PCI card, please save the result (current configuration information) of the prtpicl (1M) command.

```
# prtpicl -v > /tmp/prtpicl.pre <Return>
```

   b. Check the status of the slot.
      Subsequent examples describe extension of PCI cards to the slot "C0M01-PCI#slot02." The "C**0**M**01**-PCI#slot0**2**" indicates the cabinet number : **0**, the system board number : **1**, physical slot number : **02**.
      The relationship of the position of a PCI card and format of "Ap_Id" is shown below. (N: an integral number)

Location of PCI slot	Element	Format of "Ap_Id"
Cabinet of PRIMEPOWER 900/1500/2500	Cabinet Number : **X** System Board Number : **Y** Physical Slot Number : **ZZ**	pcipsyN:C**X**M0**Y**-PCI#slot**ZZ**
PCI / Disk Box	I/O Cabinet Number : **X** PCI / Disk Box Number : **Y** Physical Slot Number : **ZZ**	pcipsyN:R**X**B0**Y**-PCI#slot**ZZ**
other than those above	Physical Slot Number : **ZZ**	pcipsyN:PCI#slot**ZZ**

      Please confirm that the PCI slot status which extended PCI cards is "empty unconfigured" by using cfgadm (1M) command.

```
# cfgadm | grep C0M01-PCI#slot02 <Return>
pcicsy15:C0M01-PCI#slot02  unknown   empty   unconfigured   unknown
```

   c. To confirm the slot position at extension operation, blink the ALARM LED of Ap_Id displayed in

procedure 2.b.

```
# cfgadm -x led=fault,mode=blink pcipsy15:C0M01-PCI#slot02 <Return>
```

d. Please confirm that READY LED of the PCI bus slot that extends a PCI card has gone out, and extend a PCI card. This operation is performed by our customer support.

e. Please confirm that the PCI slot status which extended PCI cards is "disconnected unconfigured" by using cfgadm (1M) command.

```
# cfgadm pcipsy15:C0M01-PCI#slot02 <Return>
Ap_Id                       Type     Receptacle    Occupant      Condition
pcicsy15:C0M01-PCI#slot02  unknown  disconnected  unconfigured  unknown
```

3. Connect PCI cards
The cfgadm command is executed with the "configure" option to "Ap_Id," or the push button of the PCI bus slot is pushed. Then extended PCI card is connected.
In addition, a push button is valid only in multiple user mode.
Please confirm that the slot status is "connected configured" by using the cfgadm (1M) command after the above-mentioned procedure.

```
# cfgadm -c configure pcipsy15:C0M01-PCI#slot02 <Return>
# cfgadm pcipsy15:C0M01-PCI#slot02 <Return>
Ap_Id                   Type       Receptacle    Occupant    Condition
pcipsy15:C0M01-PCI#slot02   pci-pci/hp    connected    configured    ok
```

4. Start the hardware monitoring of the Machine Administration
Please execute the following commands in order to update configuration information, and restart the hardware monitoring of the Machine Administration.

```
# usr/sbin/FJSVmadm/postphp <Return>
```

5. Connect equipment to PCI cards
The extended PCI card is connected with network equipment by the cable.

6. Setup driver
The driver configuration is added by following operations.
   a. The interface name is investigated in order to configure drivers and high layer products.
   Please save the result of the prtcicl (1M) command, and obtain the difference information between current result and the result taken at "2.a." Then, the driver instance number for the extended PCI card is obtained.
   In the following example, since the instance number is 1, it can be determined that the interface name of the extended PCI card is "hme1."

```
# prtpicl -v > /tmp/prtpic.post <Return>
# diff /tmp/prtpicl.pre /tmp/prtpicl.post | more <Return>
        :
>   :status   okay
>   :devfs-path   /pci@8d,2000/network@1
>   :driver-name   hme
>   :binding-name   SUNW,hme
>   :bus-addr   1
>   :instance   1
>   :_class   obp-device
>   :name   network
        :
```

b. It confirms that the interface name that obtained above-mentioned procedure is an interface name that corresponds to the PCI bus slot of the added PCI card.
Please carry out this procedure, when it cannot identify relationship between an interface name and PCI bus slot by expanding two or more PCI cards especially.

```
# /usr/sbin/FJSVmadm/inst2comp hme1 <Return>

pcipsy15:C0M01-PCI#slot02
```

c. The configuration of each driver is added.
Please refer to each driver manual for details.

7. Setup applications
The configuration of extended PCI card used by high layer products (services, applications) is added.
Please refer to each high layer product manual for details.

8. Start applications
By the following operations, please start the high layer product (service, application) used by the extended PCI card.
   a. When using TCP/IP interface
   The example in the case of using TCP/IP interface is shown below.
   Furthermore, when the high layer program which is using the PCI card needs to restart, please refer to the manual of each high layer program.

```
# /usr/sbin/ifconfig hme1 plumb <Return>

# /usr/sbin/ifconfig hme1 inet 192.168.10.10 netmask + broadcast + -trailers
up <Return>
```

   b. When using FNA/SNA communication protocol over Ethernet
   When not using the redundant configuration software, it is necessary to start all interfaces of Netcompo FNA-LAN, and all high layer applications that are using Netcompo FNA-LAN.
   For details, please refer to the manual of Netcompo FNA-LAN and high layer applications.

4.2.2 Extension of PCI cards with new redundant system

This subsection describes extension procedure of a new redundant system with the following redundant configuration software.

- SynfinityLink 2.2
- PRIMECLUSTER GLS 4.0 or later

The replacement and extension of PCI cards are supported only on Redundant Line Control Function of SynfinityLink.

When using other redundant software, please refer to manuals of each product.

1. Stop the hardware monitoring of the Machine Administration
The hardware monitoring of the Machine Administration is stopped by using the following command.

```
# /usr/sbin/FJSVmadm/prephp <Return>
```

2. Add PCI cards
   a. Before extending the PCI card, please save the result (current configuration information) of the prtpicl (1M) command.

```
# prtpicl -v > /tmp/prtpicl.pre <Return>
```

   b. Check the status of the slot.
   Subsequent examples describe extension of PCI cards to the slot "R0B01-PCI#slot02." The "R**0**B0**1**-PCI#slot**02**" indicates the I/O cabinet number : **0**, the PCI / Disk Box number : **1**, physical slot number : **02**.
   In the extension of a new redundant system, although two or more PCI cards are extended, whenever you add a PCI card, please perform from procedure "2.b" to "3. Connect PCI cards."
   The relationship of the position of a PCI card and format of "Ap_Id" is shown below. (N: an integral number)

Location of PCI slot	Element	Format of "Ap_Id"

Location of PCI slot	Element	Format of "Ap_Id"
Cabinet of PRIMEPOWER 900/1500/2500	Cabinet Number : **X** System Board Number : **Y** Physical Slot Number : **ZZ**	pcipsyN:C**X**M0**Y**-PCI#slot**ZZ**
PCI / Disk Box	I/O Cabinet Number : **X** PCI / Disk Box Number : **Y** Physical Slot Number : **ZZ**	pcipsyN:R**X**B0**Y**-PCI#slot**ZZ**
other than those above	Physical Slot Number : **ZZ**	pcipsyN:PCI#slot**ZZ**

Please confirm that the PCI slot status which extended PCI cards is "empty unconfigured" by using cfgadm (1M) command.

```
# cfgadm | grep R0B01-PCI#slot02 <Return>
pcicsy18:R0B01-PCI#slot02  unknown   empty   unconfigured   unknown
```

c. To confirm the slot position at extension operation, blink the ALARM LED of Ap_Id displayed in procedure 2.b.

```
# cfgadm -x led=fault,mode=blink pcipsy18:R0B01-PCI#slot02 <Return>
```

d. Please confirm that READY LED of the PCI bus slot that extends a PCI card has gone out, and extend a PCI card. This operation is performed by our customer support.

e. Please confirm that the PCI slot status which extended PCI cards is "disconnected unconfigured" by using cfgadm (1M) command.

```
# cfgadm pcipsy15:R0B01-PCI#slot02 <Return>
Ap_Id                      Type     Receptacle    Occupant      Condition
pcicsy18:R0B01-PCI#slot02  unknown  disconnected  unconfigured  unknown
```

3. Connect PCI cards
The cfgadm command is executed with the "configure" option to "Ap_Id," or the push button of the PCI bus slot is pushed. Then a extended PCI card is connected.
In addition, a push button is valid only in multiple user mode.
Please confirm that the slot status is "connected configured" by using the cfgadm (1M) command after the above-mentioned procedure.

```
# cfgadm -c configure pcipsy18:R0B01-PCI#slot02 <Return>
# cfgadm pcipsy18:R0B01-PCI#slot02 <Return>
Ap_Id                    Type       Receptacle    Occupant    Condition
pcipsy18:R0B01-PCI#slot02   pci-pci/hp    connected    configured    ok
```

4. Start the hardware monitoring of the Machine Administration
Please execute the following command in order to update configuration information, and restart the hardware monitoring of the Machine Administration.

```
# /usr/sbin/FJSVmadm/postphp <Return>
```

5. Connect equipment to PCI cards
The extended PCI card is connected with network equipment by the cable.

6. Setup driver
The driver configuration is added by following operations.
   a. The interface name is investigated in order to configure drivers and high layer products.
   Please save the result of the prtcicl (1M) command, and obtain the difference information between current result and the result taken at "2.a." Then, the driver instance number for the extended PCI card is obtained.
   In the following example, since the instance number is 1, it can be determined that the interface name of the extended PCI card is "hme1."

```
# prtpicl -v > /tmp/prtpic.post <Return>
# diff /tmp/prtpicl.pre /tmp/prtpicl.post | more <Return>
        :
>   :status   okay
>   :devfs-path   /pci@8d,2000/network@1
>   :driver-name   hme
>   :binding-name   SUNW,hme
>   :bus-addr   1
>   :instance   1
>   :_class   obp-device
>   :name   network
        :
```

b. It confirms that the interface name that obtained above-mentioned procedure is an interface name that corresponds to the PCI bus slot of the added PCI card.
Please carry out this procedure, when it cannot identify relationship between an interface name and PCI bus slot by expanding two or more PCI cards especially.

```
# /usr/sbin/FJSVmadm/inst2comp hme1 <Return>
pcicsy18:R0B01-PCI#slot02
```

c. The configuration of each driver is added.
Please refer to each driver manual for details.

7. Setup redundant system
When establishing the redundant configuration system, it is necessary to configure the environmental configuration of the Redundant Line Control Function or the Multipath Function, and to activate the redundant configuration system.
For details of environmental configuration and activation, please refer to each function manual of SynfinityLink or PRIMECLUSTER GLS.

8. Setup applications
The configuration of high layer product (services, applications) is added.
Please refer to the manual of each high layer product for details.

9. Start applications
By following operations, please start the high layer product.
   a. When using TCP/IP interface
   The high layer product (services, applications) using the TCP/IP interface is started.
   For details of starting the high layer products, please refer to the manual of each product.

   b. When using FNA/SNA communication protocol over Ethernet
   It is necessary to start all interfaces of Netcompo FNA-LAN, and all high layer applications that are using Netcompo FNA-LAN. When using the Multipath Function, the FNA (Fujitsu Network Architecture) /SNA (System Network Architecture) communication protocol over Ethernet is valid.
   For details, please refer to the manual of Netcompo FNA-LAN and high layer applications.

4.2.3 Extension of PCI cards to redundant system

This subsection describes the procedure extending PCI cards to the existing redundant configuration system with the following redundant configuration software.

- SynfinityLink 2.2
- PRIMECLUSTER GLS 4.0 or later

The replacement and extension of PCI cards are supported only on Redundant Line Control Function of SynfinityLink.

When using other redundant software, please refer to manuals of each product.

Moreover, when extending PCI cards to the existing redundant configuration system, it is unnecessary to operate about the high layer products. Consequently, PCI cards can be extended without stopping the high layer products.

1. Stop the hardware monitoring of the Machine Administration
   The hardware monitoring of the Machine Administration is stopped by using the following command.

```
# /usr/sbin/FJSVmadm/prephp <Return>
```

2. Add PCI cards
   a. Before extending the PCI card, please save the result (current configuration information) of the prtpicl (1M) command.

```
# prtpicl -v > /tmp/prtpicl.pre <Return>
```

   b. Check the status of the slot.
   Subsequent examples describe extension of PCI cards to the slot "R0B01-PCI#slot02." The "R**0**B0**1**-PCI#slot**02**" indicates the I/O cabinet number : **0**, the PCI / Disk Box number : **1**, physical slot number : **02**.
   In the extension of a new redundant system, although two or more PCI cards are extended, whenever you add a PCI card, please perform from procedure "2.b" to "3. Connect PCI cards."
   The relationship of the position of a PCI card and format of "Ap_Id" is shown below. (N: an integral number)

Location of PCI slot	Element	Format of "Ap_Id"
Cabinet of PRIMEPOWER 900/1500/2500	Cabinet Number : **X** System Board Number : **Y** Physical Slot Number : **ZZ**	pcipsyN:C**X**M0**Y**-PCI#slot**ZZ**
PCI / Disk Box	I/O Cabinet Number : **X** PCI / Disk Box Number : **Y** Physical Slot Number : **ZZ**	pcipsyN:R**X**B0**Y**-PCI#slot**ZZ**
other than those above	Physical Slot Number : **ZZ**	pcipsyN:PCI#slot**ZZ**

   Please confirm that the PCI slot status which extended PCI cards is "empty unconfigured" by using cfgadm (1M) command.

```
# cfgadm | grep R0B01-PCI#slot02 <Return>
pcicsy18:R0B01-PCI#slot02  unknown   empty   unconfigured   unknown
```

   c. To confirm the slot position at extension operation, blink the ALARM LED of Ap_Id displayed in procedure 2.b.

```
# cfgadm -x led=fault,mode=blink pcipsy18:R0B01-PCI#slot02 <Return>
```

   d. Please confirm that READY LED of the PCI bus slot that extends a PCI card has gone out, and extend a PCI card. This operation is performed by our customer support.

   e. Please confirm that the PCI slot status which extended PCI cards is "disconnected unconfigured" by using cfgadm (1M) command.

```
# cfgadm pcipsy18:R0B01-PCI#slot02 <Return>
Ap_Id                       Type     Receptacle    Occupant      Condition
pcicsy18:R0B01-PCI#slot02  unknown  disconnected  unconfigured  unknown
```

3. Connect PCI cards
   The cfgadm command is executed with the "configure" option to "Ap_Id," or the push button of the PCI bus slot is pushed. Then a extended PCI card is connected.
   In addition, a push button is valid only in multiple user mode.
   Please confirm that the slot status is "connected configured" by using the cfgadm (1M) command after the above-mentioned procedure.

```
# cfgadm -c configure pcipsy18:R0B01-PCI#slot02 <Return>
# cfgadm pcipsy18:R0B01-PCI#slot02 <Return>
```

```
Ap_Id                    Type       Receptacle    Occupant    Condition
pcipsy18:R0B01-PCI#slot02   pci-pci/hp    connected    configured    ok
```

4. Start the hardware monitoring of the Machine Administration
   Please execute the following command in order to update configuration information, and restart the hardware monitoring of the Machine Administration.

```
# /usr/sbin/FJSVmadm/postphp <Return>
```

5. Connect equipment to PCI cards
   The extended PCI card is connected with network equipment by the cable.

6. Setup driver
   The driver configuration is added by following operations.
   a. The interface name is investigated in order to configure drivers and high layer products.
      Please save the result of the prtcicl (1M) command, and obtain the difference information between current result and the result taken at "2.a." Then, the driver instance number for the extended PCI card is obtained.
      In the following example, since the instance number is 1, it can be determined that the interface name of the extended PCI card is "hme1."

```
# prtpicl -v > /tmp/prtpic.post <Return>
# diff /tmp/prtpicl.pre /tmp/prtpicl.post | more <Return>
        :
>   :status   okay
>   :devfs-path   /pci@8d,2000/network@1
>   :driver-name   hme
>   :binding-name   SUNW,hme
>   :bus-addr   1
>   :instance   1
>   :_class   obp-device
>   :name   network
        :
```

   b. It confirms that the interface name that obtained above-mentioned procedure is an interface name that corresponds to the PCI bus slot of the added PCI card.
      Please carry out this procedure, when it cannot identify relationship between an interface name and PCI bus slot by expanding two or more PCI cards especially.

```
# /usr/sbin/FJSVmadm/inst2comp hme1 <Return>
pcipsy18:R0B01-PCI#slot02
```

   c. The configuration of each driver is added.
      Please refer to each driver manual for details.

7. Connect to redundant system
   Please execute following commands in order to connect the extended PCI card to the existing redundant configuration system.

   Notes:
   On the NIC switching mode in Redundant Line Control Function, it is impossible to extend a new interface to the existing redundant configuration system. Moreover, when extending an interface on the GS/SURE linkage mode, it is necessary to extend an interface after deactivating the virtual interface (e.g. sha0) used by the GS/SURE linkage mode.

```
Fast switching mode in Redundant Line Control Function
```

```
# /usr/sbin/ifconfig hme2 plumb <Return>
# /usr/sbin/ifconfig hme2 192.168.20.10 netmask + broadcast + -trailers up <Return>


GS/SURE linkage mode in Redundant Line Control Function

# /opt/FJSVhanet/usr/sbin/stphanet -n sha0 <Return>
# /opt/FJSVhanet/usr/sbin/hanetconfig create -n sha2 -m n -i 192.168.20.10 -t hme2
<Return>
# /opt/FJSVhanet/usr/sbin/hanetnic modify -n sha0 -t sha2,sha3,sha4 <Return>
# /opt/FJSVhanet/usr/sbin/strhanet -n sha0 <Return>


Multipath Function

# /usr/opt/FJSViomp/bin/iompadm -c mpnet add mpnet0 hme2 <Return>
# /usr/opt/FJSViomp/bin/iompadm -c mpnet start mpnet0 hme2 <Return>
```

8. Switch the redundant path
   Please switch an online communication path for extended communication path if needed. Please refer to manuals of each redundant configuration software about how to switch a communication path.

Chapter 5 Notes, Restrictions

This chapter describes notes and restrictions in case a PCI card exchanges or extends by using PCI Hot Plug feature.

5.1 Notes

This section describes notes when exchanging and extending PCI cards by using PCI Hot Plug feature.

5.1.1 Network driver configuration changing

When configuring parameters in driver.conf(4) file and replacing the PCI card by using the PCI Hot Plug feature, parameters which configured in driver.conf(4) file are not valid except for following conditions.

- In the Solaris 9 OE system by which only one PCI card that is supported by one driver is mounted, please execute the following command after the replacement procedure "6. Connect the PCI card," when replacing the PCI card by using the PCI Hot Plug feature. Consequently, parameters which configured in driver.conf(4) file are reflected.

  # **update_drv** *driver_name* <Return>

- In the configuration of the driver.conf(4) file, when not specifying "name", "parent", "unit-address" parameters, configured parameters are reflected. However, all the PCI cards that the driver supports become the same configuration.

When not corresponding to the above-mentioned conditions, and configuring parameters after the PCI card is replaced or extended by using PCI Hot Plug feature, please refer to the solution of each PCI card show below.

Product ID (Functions)	Software of driver	Resolution
X1033A-U X1033A-X (10/100BASE-T)	Solaris OE(hme)	Please make a setting by the ndd command. In case of reflecting the setting of /kernel/drv/hme.conf file using *name, parent* and *unit-address* parameters, please reboot a system.
PW008FE1 (10/100BASE-T)	FUJITSU PCI 10/100 Ethernet 1.0 or later (enclosed with a PCI card)	Please make a setting by the ndd command. In case of reflecting the setting of /kernel/drv/fjfe.conf file using *name, parent* and *unit-address* parameters, please reboot a system.
PW008QE1 (10/100BASE-T x 4)	FUJITSU PCI Quad 10/100 Ethernet 1.0 or later (enclosed with a PCI card)	Please make a setting by the ndd command. In case of reflecting the setting of /kernel/drv/fjqe.conf file using *name, parent* and *unit-address* parameters, please reboot a system.
PW008QE2 (10/100BASE-T x 4)	FUJITSU PCI Quad 10/100 Ethernet 1.1 or later (enclosed with a PCI card)	
PP028GE1U PP028GE1X (1000BASE-SX)	FUJITSU PCI Gigabit Ethernet 1.4.1 or later (enclosed with a PCI card)	ndd command is not supported. In case of reflecting the setting of /kernel/drv/fjge.conf file using *name, parent* and *unit-address* parameters, please reboot a system.
PW008GE1U PW008GE1X (1000BASE-T)		
PW008GE4 (1000BASE-SX)	FUJITSU PCI Gigabit Ethernet 2.0 or later (enclosed with a PCI card)	Please make a setting by the ndd command. In case of reflecting the setting of /platform/sun4us/kernel/drv/fjgi.conf file using *name, parent* and *unit-address*, please reboot a system.
PW008GE5 (10/100/1000BASE-T)		

5.1.2 Using tape device while PCI Hot Plug operation

By using the PCI Hot Plug feature, a job which uses tape devices may be terminated abnormally.
While using the PCI Hot Plug feature, avoid using tape devices.

5.1.3 Others

- When expanding PCI cards by using PCI Hot Plug feature, it is necessary to install the device driver (case of file devices) and the driver for using the added PCI card.
- When extending PCI cards, each driver consumes memory capacity to control PCI cards.
  Please refer to the installation guide attached to each PCI card about memory capacity required in order to extend PCI cards, and installation guide of the redundant configuration software about memory capacity required.

5.2 Restrictions

This section describes restrictions when exchanging and extending PCI cards by using PCI Hot Plug feature.

- Please do not exchange or expand PCI cards by using PCI Hot Plug feature during executing Dynamic Reconfiguration (DR).